High-Level Optimization of Performance and Power in Very Deep Sub-Micron Interconnects

Vom Fachbereich 18
Elektrotechnik und Informationstechnik
der Technischen Universität Darmstadt
zur Erlangung der Würde eines
Doktor–Ingenieurs (Dr.-Ing.)
genehmigte Dissertation

von

Dipl.-Ing.
Tudor A. Murgan
geboren am 7. Februar 1978
in Bukarest, Rumänien

Referent:	Prof. Dr. Dr. h. c. mult. Manfred Glesner *Technische Universität Darmstadt*
Korreferent:	Prof. Dr. Mircea R. Stan *University of Virginia, Charlottesville*
Tag der Einreichung:	18.07.2006
Tag der mündlichen Prüfung:	06.10.2006

D17

Darmstädter Dissertationen

Familiei mele…

dedic
şi mă dedic.

"Arbor invers am rămas, rupt din sferă
cu sfera aceasta aidoma, geamănă…
Şi totul îmi pare ştiut, dar nimica
din ce ştiu cu ce este nu se aseamănă…"

Nichita Stănescu

"Nihil tam bene dictum quod non possit depravari."
Abælardus

Preface

This dissertation is the consequence of the work as a teaching and research assistant at the Institute of Microelectronic Systems of the Darmstadt University of Technology. I would like to sincerely thank my *Doktorvater*, Prof. Manfred Glesner, not only for his kind advice and guidance that made this thesis possible, but also for giving me the opportunity to be involved in several teaching activities and consistent research projects funded by important companies and scientific foundations.

I also express my gratitude towards Prof. Mircea Stan from the University of Virginia in Charlottesville, who kindly accepted to act as a reviewer for this thesis. His comments and observations have been very valuable for improving the quality of the work, and as a result of our discussions, new ideas emerged.

Furthermore, I am indebted to Prof. Jürgen Adamy, Prof. Alex Gershman, and Prof. Jürgen Stenzel as members of the examination committee. I would especially like to thank Prof. Gershman for the fruitful discussions around the topic of the dissertation.

This work could not have been accomplished smiling without a good atmosphere at the working place. For the provided pleasant environment, I would like to express special thanks to all colleagues at the institute, with whom I had the pleasure of carrying out interesting research projects, producing dozens of papers and reports, sharing various teaching activities, and solving numerous stringent administrative issues. The friendly help and support of Petru Bacinschi, Oana Cobianu, Andre Guntoro, Heiko Hinkelmann, Hans-Peter Keil, Massoud Momeni, and Hao Wang permitted me – especially during the last months – to concentrate on writing the final manuscript and on preparing the exam. Moreover, I will never forget the late-night spicy discussions, spanning from scientific and projects-related to political and historical issues, with the colleagues of the same "generation" (or not - *sic!*), with whom I spent at least four years at the institute: Leandro Soares Indrusiak, Abdulfattah Obeid, Sujan Pandey, and Oliver Soffke. In this context, I would especially like to mention the collaboration with Mihail Petrov, with whom I shared for many years office, projects, reports, presentations, meetings, deadlines, worries, and watering of plants.

I am also greatly indebted to the older, former, and external colleagues who shared, on various occasions, their experience regarding a multitude of issues like writing papers and project proposals, finalizing project reports, and understanding of the given facts (f.i. earth rotation): Jürgen Deicke, Alberto García Ortiz, Thomas Hollstein, Lukusa

Kabulepa, Ralf Ludewig, Octavian Mitrea, Juan Jesús Ocampo Hidalgo, Thilo Pionteck, Matthias Rychetsky, Clemens Schlachta, Burkart Voß, Heiko Zimmer, Peter Zipf, Klaus Koch, Márcio Kreutz, José Palma, and Gilles Sassatelli.

Further, I would like to thank the former students I have been able to work with throughout these years, especially Ismail Deflaoui, Mateusz Majer, Oliver Mitea, and Alexander Werth. Thanks also to the competent and good-humored assistance of the system administrator, Andreas Schmidt, as well as to the friendly support of our secretaries, Silvia Hermann and Irmgard Wackermann.

In particular, I would like to acknowledge the exceptional support received from my colleagues Alberto García Ortiz, Ralf Ludewig, Massoud Momeni, and Petru Bacinschi. Alberto and Ralf have been not only my first research mates, but also patient "initiators" and "real-time online debuggers". The comments and constructive criticism of Alberto, Massoud, Petru, and Ralf provided a solid foundation for finding the "red line", carrying out the research, and clarifying several key issues. My good old friend Nicolae Statu gave very generously of his time to read the manuscript and provide essential feedback regarding the language used.

During the last five years, I had the chance to visit many conferences, institutions, and universities around the world and thus, to meet many researchers and design engineers. Those more constant or punctual interactions contributed significantly to finalizing and improving the results of this work. In this context, I would like to mention the discussions with Prof. Enrico Macii, Prof. Mircea Stan, Dr. Vladimir Zolotov, Prof. Luis Miguel Silveira, Prof. Radu Mărculescu, Prof. Ricardo Reis, to name only a few.

My stay in Darmstadt would have not been possible without the professional and friendly support received from several people from the "Politehnica" during and after my studies in Bucharest. Therefore, I would like to gratefully acknowledge the continuous encouragement of Prof. Anca Manuela Manolescu, Prof. Anton Manolescu, Prof. Felicia Ionescu, and Prof. Radu Dogaru.

Old and new friends allowed us a seamless adaptation to the living in a new city and a new country. For the wonderful time spent together, many thanks to Giuliana and Leandro, Alina and Radu, Blanca and Alberto, Monica and Nicolae, Oana and Tavi, Mişu, Ralf, Jayjay, Claudia and Burkart, Andrea and Matthias, and all those fantastic friends which we have been meeting only roughly twice a year when traveling to Bucharest.

Last but not least, I wish to express my heartfelt gratitude to my entire family for all their efforts, for the received education and opportunities, for their unconditional continuous support, encouragement, trust, and friendship. The deepest thoughts to my inspiring wife, Ilina.

Darmstadt, November 2006

Abstract

Interconnect analysis and optimization at high levels of abstraction is extremely attractive since it offers a much larger room for improvement than optimization at lower levels. The objective of this thesis is to optimize performance and power consumption in interconnect structures at high levels of abstraction. For this purpose, efficient high-level models for delay and power consumption in very deep sub-micron interconnects are developed and employed for constructing and evaluating different low power and throughput improving signal encoding schemes. Moreover, in order to achieve an even higher efficiency, coding is combined with lower level techniques like spacing, shielding, and buffer planning.

In order to construct and evaluate encoding schemes at high levels of abstraction, two conceptually different issues must be solved. On the one hand, bit-level characteristics of the data transmitted over the interconnect structures need to be extracted during system-level design and architecture specification. On the other hand, important interconnect-related very deep sub-micron effects have to be incorporated into high-level models as well. Delay models able to predict the line delay for each set of input patterns (and not only for the worst case) are required in order to develop and evaluate coding schemes tailored for performance improvement. An essential contribution of this work is the development of a pattern-dependent delay model. The essence of the so-called extended linear delay (ELD) model is to incorporate the effects of all possible input patterns in buses exhibiting not only inter-wire capacitance, but also inductive effects which are in general more difficult to predict and more daunting because of their long-range nature. Further, the described power macromodel shows that in order to decrease dynamic power consumption at high levels of abstraction, one has to reduce not only the self transition activity but also the so-called coupling transition activity responsible for charging and discharging the inter-wire capacitances in a bus.

The abovementioned models are employed in order to construct and evaluate several low-power and throughput improving codes. Based on the observation that the bit-level transition activity in typical DSP applications can be accurately described by two breakpoints, several simple yet very efficient hybrid codes are constructed. Those codes combine non-redundant and redundant schemes in such a way that the total self and coupling transition activity are significantly decreased. Moreover, maximum achievable limits are derived, which show the effectiveness of the developed codes. Further, several

low-complex codes are proposed that improve bus performance by avoiding a certain set of input patterns. In this context, fundamental limits and bounds are derived for state and transition coding, respectively. Coding is also compared and combined with low-level interconnect optimization techniques like spacing and shielding. The problem of simultaneously addressing coding-based power reduction and performance improvement is introduced and analyzed.

Finally, an interconnect-centric design flow is presented that integrates signal encoding for power and performance optimization. Signal encoding schemes can be constructed at high levels of abstraction while analyzing the data that is transmitted through the interconnect system. After interconnect planning and synthesis, when exact information regarding layout and routing optimization is available, codes can be refined based on the specific wire topology. Moreover, in order to prove the large optimization opportunities available at high levels, a simultaneous buffer insertion and placement algorithm is developed. In this context, coding for throughput is appended to the developed algorithm, and it is shown that performance and/or power consumption can be thus further improved.

Kurzfassung

Die Analyse und Optimierung von Verbindungsstrukturen in integrierten Schaltungen auf hohen Abstraktionsebenen ist äußerst attraktiv, da diese im Vergleich zu niedrigeren Ebenen deutlich mehr Verbesserungsmöglichkeiten anbieten. Ziel dieser Dissertation ist es Leistungsverbrauch und Performanz in Verbindungsstrukturen auf hohen Abstraktionsniveaus zu optimieren. Für diesen Zweck werden effiziente Modelle für Signalverzögerung und Leistungsverbrauch in sub-100 nm (*very deep sub-micron*) Verbindungsstrukturen erstellt und anschließend auf die Entwicklung und Bewertung verschiedenartiger Kodierungsmethoden angewendet, die den Leistungsverbrauch verringern und den Datendurchsatz verbessern. Um eine noch größere Effizienz zu erzielen, werden Kodierungsschemen mit Methoden wie Abstandvergrößerung, Abschirmung und Einfügen von Leitungstreibern kombiniert, die üblicherweise auf den unteren Abstraktionsebenen angewandt werden.

Um wirksame Kodierungsmethoden entwickeln und analysieren zu können, müssen zwei grundlegend verschiedene Probleme gelöst werden. Auf der einen Seite ist für die Entwicklung der Kodierungsmethoden notwendig, auf der System- und Architekturebene bedeutende Eigenschaften der zu sendenden Daten zu extrahieren. Auf der anderen Seite ist für deren korrekte Bewertung erforderlich, wesentliche technologiebedingte Effekte in Makromodellen auf höheren Abstraktionsebenen einzubinden. Die Evaluierung von Kodierungstechniken, die den Durchsatz erhöhen, kann nur dann erfolgen, wenn die verwendeten Verzögerungsmodelle die von allen möglichen Eingangstransitionen erzeugten Verzögerungen vorhersagen können und nicht nur die ungünstigsten (*worst case*) Fälle betrachten. In diesem Zusammenhang wird in dieser Arbeit ein transitionsabhängiges Verzögerungsmodell entwickelt, das sowohl kurzreichende kapazitive Kopplungen als auch weitreichende und somit unübersichtlichere induktive Effekte beücksichtigt. Des Weiteren wird auch ein Makromodell für den Leistungsverbrauch beschrieben. Dieses Makromodell zeigt im Wesentlichen, dass sich die Optimierung des dynamischen Leistungsverbrauchs auf hohen Abstraktionsebenen auf die Verringerung sowohl der Eigenschaltaktivität als auch der sogenannten Koppelschaltaktivität reduziert, die für das Umladen der Koppelkapazitäten verantwortlich ist.

Im Laufe der Arbeit werden die oben genannten Modelle für die Entwicklung und Bewertung von verschiedenartigen und optimierten Kodierungsmethoden verwendet. Basierend auf der Beobachtung, dass die Schaltaktivität in typischen Sig-

nalverarbeitungsarchitekturen mittels zwei sogenannter Grenzpunkte modelliert werden kann, werden verschiedene hybride Kodierungstechniken entwickelt, die nichtredundante und redundante Methoden kombinieren, sodass die Eigen- und Koppelschaltaktivität stark reduziert werden. Ferner werden theoretische Schranken für die Reduzierung der Schaltaktivität abgeleitet, um die Effektivität der vorgeschlagenen Kodierungstechniken nachzuweisen. Außerdem werden mehrere durchsatzerhöhende Kodierungsmethoden entwickelt, in denen eine bestimmte Menge von Eingangstransitionen ungültig gemacht wird. In diesem Zusammenhang werden sowohl grundsätzliche Schranken für Zustands- und Transitions-Kodierung berechnet als auch Vergleiche mit Verzögerungsoptimierungsmethoden wie Abstandvergrößerung und Abschirmung durchgeführt, die auf niedrigeren Abstraktionsebenen angewandt werden. Es wird gezeigt, dass durch das Zusammenlegen von Kodierung und solcher Methoden eine verbesserte Effizienz erreicht werden kann.

Schließlich wird eine Entwurfsmethodik für integrierte Schaltungen und Systeme beschrieben, in deren Mittelpunkt die Optimierung von Verbindungsstrukturen steht. Kodierungsschemen können im Wesentlichen während der ersten Entwurfsphasen entwickelt und analysiert werden, da die bedeutendsten Eigenschaften der gesendeten Daten zur gleichen Zeit extrahiert werden können. Nach der Planung und Synthese der Verbindungsstrukturen und der dazugehörigen Kodierungen sind exakte Details zu der endgültigen Geometrie der Verbindungsstrukturen bekannt. Folglich können Kodierungen weiter in einer leitungsspezifischen Weise verfeinert werden. Darüber hinaus wird ein Algorithmus entwickelt, der die Plazierung und das Einfügen von Leitungstreibern gleichzeitig durchführt, um somit die beachtlichen Optimierungsmöglichkeiten hervorzuheben, die auf hohen Abstraktionsebenen vorhanden sind. Die Erweiterung des entwickelten Algorithmus mit Kodierungsmethoden erlaubt eine Verbesserung des Durchsatzes und/oder des Leistungsverbrauchs.

Table of Contents

List of Abbreviations

APBI	Adaptive Partial Bus Invert
APOEBI	Adaptive Partial Odd/Even Bus Invert
ARMA	Auto-Regressive Moving Average
AWE	Asymptotic Waveform Evaluation
BI	Bus Invert
BITS	Bus Invert and Transition Signaling
BISWS	Buffer Insertion/Sizing and Wire Sizing
BSIM	Berkeley Short-channel IGFET Model
CAD	Computer-Aided Design
CMOS	Complementary Metal Oxide Semiconductor
D-RLL	Differential Run Length Limited
D2M	Delay Two Moments
DIBL	Drain-Induced Barrier Lowering
DSM	Deep Sub-Micron
DSP	Digital Signal Processing
ELD	Extended Linear Delay
ETAM++	Extended Transition Activity Measure
FFT	Fast Fourier Transform
FPGA	Field Programmable Gate Array
GALS	Globally Asynchronous Locally Asynchronous
hihrTS	Half-identity half-reverse and Transition Signaling
HP	Half-Perimeter
IGFET	Insulated-Gate Field-Effect Transistor
ITRS	International Technology Roadmap for Semiconductors
LSB	Least Significant Bit
LWC	Limited-Weight Codes
M-RLL	Modified Run Length Limited
MOS	Metal Oxide Semiconductor
MOR	Model Order Reduction
MSB	Most Significant Bit

NMOS	N-Type MOS
NoC	Network-on-Chip
OEBI	Odd/Even Bus Invert
OFDM	Orthogonal Frequency Division Multiplex
QAM	Quadrature Amplitude Modulation
PMOS	P-Type MOS
PBI	Partial Bus Invert
PBIC	Partial Bus Invert Coupling
PBIH	Partial Bus Invert Hamming
PCA	Principal Component Analysis
PCB	Printed Circuit Board
PEEC	Partial Element Equivalent Circuit
PDF	Probability Density Function
PDP	Power-delay Product
POEBI	Partial Odd/Even Bus Invert
PRIMO	Probability Interpretation of Moments for Delay Calculation
PUL	Per Unit Length (Partial Unit Length)
rPEEC	Retarded PEEC
RAT	Required Arrival Time
RF	Radio Frequency
RLL	Run Length Limited
RTL	Register Transfer Level
SA	Simulated Annealing
SERT	Steiner Elmore Routing Tree
SINO	Simultaneous Shield Insertion and Net Ordering
SPICE	Simulation Program with Integrated Circuit Emphasis
TEM	Transverse Electromagnetic Mode
TPC	Transition Pattern Coding
UDSM	Ultra Deep Sub-Micron
VDSM	Very Deep Sub-Micron
WC	Worst Case
WED	Weibull-based Delay
XOR	Exclusive Or Operator

List of Symbols

C_s (C_g)	Self (Ground) capacitance
C_c	Coupling capacitance
C_{eff}	Effective capacitance
h	Wire height
l	Wire length
p	Wire pitch
s	Wire spacing
t	Wire thickness
w	Wire width
N	Bus width
S	Number of segments
Z_d	Driver impedance
Z_l	Characteristic impedance of the line
Z_o	Output impedance
$t_{s\,i}$	Bit self transition activity
$t_{c\,i}$	Bit coupling transition activity
$t_{eq\,i}$	Bit equivalent transition activity
$t_{eq,m}$	Mean equivalent transition activity
t_m	Self activity in the MSB
dt_m	Self activity excess in the MSB
T_s	Total self transition activity
T_c	Total coupling transition activity
T_{eq}	Total equivalent transition activity
$\theta_{c\,ij}$	Inter-wire coupling activity
T_{Wc}	Weighted total coupling transition activity
σ	Standard deviation
σ_n	Normalized standard deviation
ρ	Correlation
BP_0, BP_1	Breakpoints 0 and 1, respectively
ΔB_k	Correction factor for breakpoint k

t_r	Rise time
t_d	Delay time
t_p	Propagation time
$t_{p,seg}$	Propagation time in one segment
f_s	Significant frequency
ξ	Damping factor
τ_{RC}	RC time constant
τ_{LC}	LC time constant
τ_{Di}	Elmore delay in node i
T_{ck}	Clock period
t_{di}	Delay in node i
δ_k	Delay in line k
δ_m	Mean delay
ψ_j	Generic (j-th) PUL parameter
b_k^+, b_k^-	Current and previous values on line k
Δb_k	Transition in line k
κ, κ_{ij}	Bus aspect factor
α_{ij}	ELD model coefficients
$\mu_k^{\alpha_{ij}}$	k-th moment of the random variable α_{ij}
Δ_k	Delay class $k-1$
S_{ind}	Cumulative (inductive) influence of higher order aggressors
η	Max. (inductive) cumulative effect of higher order aggressors
ζ_b	Bit rate reduction factor
ζ_s	Speed increasing factor
ζ_t	Throughput increase rate
A_G	Adjacency graph matrix
G	Generating (recurrence) matrix
$\rho(G)$	Spectrum of matrix G
λ_k	k-th eigenvalue
ϱ	Constrained channel capacity
H	Entropy
F_n	Fibonacci number
φ	Golden ratio

List of Tables

List of Figures

Chapter 1

Introduction and Overview

Contents

> *"Reduced cost is one of the big attractions of integrated electronics, and the cost advantage continues to increase as the technology evolves toward the production of larger and larger circuit functions on a single semiconductor substrate... The complexity for minimum component costs has increased at a rate of roughly a factor of two per year. Certainly over the short term, this rate can be expected to continue, if not to increase. Over the long term, the rate of increase is a bit more uncertain, although there is no reason to believe it will not remain nearly constant for at least 10 years..."*

The above-quoted observation made in 1965 by Gordon Moore [118], that the number of transistors per unit area on integrated circuits had doubled every year since the integrated circuit was invented, led to a prediction which the more widely it became accepted, the more it served as a goal for the entire semiconductor industry. Moore predicted that due to the attractiveness of continuous design cost reduction, this trend would continue for the foreseeable future. In the immediately following years, the pace of miniaturization slowed down a bit as it has doubled approximately every 18 to 24 months [119]. Nevertheless, most experts – including Moore himself (*"Another decade is probably straightforward... There is certainly no end to creativity..."*) – expect the law to hold for at least another two decades, even though its growth rate might slow down slightly [120]. While this time horizon is not impossible, it does not come without serious manufacturing and design challenges.

The fundamental implication of Moore's law is that with every new technological node, more, smaller, cheaper, and faster devices can be integrated in the same die area.

Moreover, with increasing die sizes, the total number of integrated transistors grew at an even faster rate, leading thus not only to more computing performance, but also to an increased added functionality and system complexity. As silicon-based components and platform ingredients gain in performance, they become exponentially cheaper to produce, and therefore more plentiful, more powerful, and more seamlessly integrated in our daily lives [65].

1.1 Motivation

Throughout the last decades, several potential physical or manufacturing barriers and design-related roadblocks have been prefigured. For instance, many argued that the 0.35 µm technology would represent the physical limit for photolithography. Nevertheless, even though the wavelength of light is approximately equal to that value, further improvements and advances permitted scaling below that barrier. At this point, the term *deep sub-micron (DSM)* emerged in order to accentuate that another scaling limit had been successfully passed [62]. From this point on, the device behavior was characterized by a growing number of so-called *short-channel effects* and the interconnects started to pose severe difficulties in timing closure due to higher RC (resistance-capacitance) delays, propagation times, inter-wire coupling and noise injection. Furthermore, the increasing resistance in the power distribution lines began to generate voltage drops in the power grid – the so-called IR drops. Copper was introduced instead of aluminium in order to cope with the increased line resistance as well as with the associated reliability issues due to metal migration caused by high currents. Nevertheless, the introduction of copper merely delayed those problems for a couple of technological nodes rather than solving them. All the abovementioned issues are collectively known as deep sub-micron effects. Even though the aforementioned problems have been in general only partly solved, scaling continued at basically the same rate and the 45 nm and 30 nm technologies are currently in sight. The era with technologies going under 100 nm is commonly referred to as *very deep sub-micron (VDSM)* or *ultra deep sub-micron (UDSM)*.

Some of the major challenges in integrated circuits that use nanoscale transistors are increasing process parameter variations and leakage currents. As a result of variation and leakage, the margins available for predictive design are becoming harder to achieve and additionally such systems dissipate considerable power even when not switching. Furthermore, increasing power consumption and thus heat dissipation pose a tremendous pressure not only on the overall system reliability but also on packaging due to severe thermal requirements. Other serious issues related to VDSM devices are velocity saturation, thin-oxide, random doping fluctuations, drain-induced barrier lowering, and hot-carrier effects [62, 141].

By cramming more components on a single chip and with augmenting die sizes, the routing requirements increased. As the routing capacity of the existing layers was ex-

Tab. 1.1: Major very deep sub-micron device and interconnect issues (after [62] and [66])

VDSM devices	VDSM interconnects
Short-channel effects	RC and RLC delays
Velocity saturation	IR drops
Thin-oxide (tunneling/breakdown)	$L\frac{di}{dt}$
Subthreshold current	Capacitive and Inductive Coupling (Crosstalk)
Drain-induced barrier lowering	Electromigration
Hot-carrier effects	Antenna Effects

hausted, other layers have been added on top of the existing ones. Consequently, the length of global and intermediate interconnects increased and the wire capacitance could not be neglected anymore. As explained in **Chap**. 2, the resistance of global and semi-global lines increased dramatically, causing significant wire delay. In order to cope with augmenting interconnect delay, the wire aspect ratio had to be increased. The improved line resistivity came, however, at the expense of larger capacitive crosstalk between neighboring wires. Moreover, with decreasing rise and fall times, i.e. growing frequency spectra of on-chip signals, non-negligible inductive effects appeared. Because of their long-range nature, inductive effects – even though less frequent yet – are more difficult to predict and control and therefore generally even more dampening than the capacitive ones, which are restricted to a short range. Further underlying problems are electromigration and antenna effects [62, 141]. The most important VDSM effects related to device and interconnect scaling are listed in **Tab**. 1.1.

Buffer insertion is a very effective and probably the most popular method to reduce interconnect delay and lately also crosstalk, by breaking long wires into shorter ones and inserting signal repeating gates. Due to the fact that control of power dissipation and density is becoming in many ways more daunting than timing closure [24], the importance of estimating and optimizing power consumption at early design stages is steadily increasing. Traditionally, the inserted buffers barely influenced the total area and power consumption of a system. However, repeaters are reported to become a problem at both chip- and block-level, as the percentage of total (local and global) repeaters in a design is projected to reach 35 % by the 45 nm technology node and even 70 % by the 32 nm node [165]. This means that buffers will eventually be responsible for the majority of the die area and total static power consumption (leakage-induced). The dynamic component of power consumption will be primarily determined by the total interconnect structure, i.e. switching capacitances of wires and repeaters. Such an explosion in repeater number will finally have a profound impact on the entire design flow, since issues like minimizing area and power consumption in buffered interconnects need to be tackled when the largest optimization opportunities are available, that is during the very early stages of the design flow [24, 30, 32, 165].

Two fundamental observations can be made with regard to interconnect structures. On the one hand, wires cannot be treated anymore as an afterthought, and on the other hand, interconnect structures decisively influence not only the overall system performance but also the total power consumption of the system. Given this increasingly dominant importance of interconnects, design flows have to be adapted to accommodate interconnect analysis, synthesis, and optimization methods at every level of abstraction, especially at higher ones. The envisaged emphasis on interconnects requires the integration of both novel specific algorithms and fundamentally new methodologies with the associated tool flows. In the 2005 SIA International Technology Roadmap for Semiconductors, both technology- and design-related interconnect issues have been identified as major challenges especially because *"traditional interconnect scaling will no longer satisfy performance requirements"* [66, 67]. In order for those issues not to become possible showstoppers for a successful continuation of Moore's law, several problems must be solved: defining and finding material solutions beyond copper and low-K, accelerated design, novel packaging techniques, unconventional interconnects, advanced interconnect-centric design, accurate and efficient modeling of VDSM effects.

1.2 Research Scope and Objectives

Interconnect analysis and optimization at high levels of abstraction is extremely attractive since it offers a much larger room for improvement than optimization at lower levels. The goal of the present dissertation is to optimize or at least improve performance and power consumption in interconnect structures at high levels of abstraction. For this purpose, efficient high-level models for delay and power consumption in very deep sub-micron interconnects are developed and employed for constructing and evaluating different low power and throughput improving signal encoding schemes.

For efficiently estimating and improving delay, accurate pattern-dependent delay models are required that are able to model the effect of each input pattern rather than only that of the worst case patterns, and the models must incorporate not also capacitive coupling but also inductive effects. Moreover, in order to efficiently decrease the dynamic power consumption, the transition activity must be reduced. In buses with inter-wire capacitance, it is not sufficient to decrease the so-called self transition activity, i.e. the transition activity related to charging and discharging the ground capacitances. Since the coupling capacitances are comparable to the ground capacitances or may even dominate them in VDSM technologies, the so-called coupling transition activity responsible for charging and discharging the inter-wire capacitances must be decreased rather than only the self activity.

This thesis advocates a paradigm shift in current integrated circuits design flows towards a more careful optimization at multiple levels of abstraction and especially at the higher ones of interconnect structures, i.e. wires and the associated clocked or non-

clocked buffers. Among others, an effective interconnect-centric design flow must integrate signal encoding schemes and this type of optimization must be performed in strong relation with an accurate high-level estimation of data characteristics and lower level interconnect optimization techniques like buffer insertion and planning, wire sizing, shaping, spacing, shielding, and splitting.

1.3 Thesis Outline

This thesis consists of three main parts. The introductory part comprises motivation, problem formulation, required background, and state-of-the-art. Secondly, the core of the work presents the introduced macromodels, the developed coding schemes, and the determined bounds for signal encoding. In order to achieve a higher efficiency, coding is combined with lower level techniques. The thesis ends with a description of the design flow integration of the proposed techniques and several concluding remarks.

Part I **Chap**. 2 and **Chap**. 3 represent the background required for constructing the proposed coding-based methodology. The goal of **Chap**. 2 is to prove the strong interrelation between signal activity and input patterns effects on one side, and crosstalk, performance, and power on the other side. For this purpose, several interconnect and driver models are compared based on the most important modeled physical effects. Further, extensive simulations are carried out that show the strong influence of input patterns on the aforementioned design metrics. **Chap**. 3 discusses a multitude of optimization techniques available at different levels of abstraction. The focus lies thereby more on high-level optimization techniques. In this context an already proposed interconnect-centric design flow and existing signal encoding schemes are reviewed.

Part II **Chap**. 4, **Chap**. 5, and **Chap**. 6 represent the core of this thesis. The proposed power and pattern-dependent delay macromodels are constructed in **Chap**. 4. The essence of the power macromodel is that at high levels of abstraction the problem of reducing dynamic power consumption by means of signal encoding schemes is equivalent to decreasing the equivalent transition activity. The power macromodel encompasses also dynamic delay effects that appear in buffered interconnects and also process variations typical for VDSM technologies. Moreover, the notion of weighted transition activity is introduced in order to extend its applicability also to unsymmetrical buses. The delay model allows the estimation of the line delay produced by each input pattern also in the case of inductive lines. The aforementioned models are employed to construct and analyze power and throughput improving coding schemes. In **Chap**. 5, the bit-level activity in typical DSP architectures is analyzed. Non-redundant codes are combined with redundant ones in order to decrease the self and coupling transition activities. As a result of the observation

that the least significant bits (LSBs) can be accurately modeled as uncorrelated uniformly distributed data, and that the most significant bits (MSBs) exhibit a significant temporal and spatial correlation, hybrid coding schemes based on classical bus invert schemes are constructed. The main idea of the codes is to employ a non-redundant code similar to a Gray mapping for the MSBs and bus invert schemes for the LSBs. Moreover, the breakpoints (delimiters of the MSB and LSB regions) can be estimated on-line by monitoring the transition activity in a selected set of lines. In order to show the effectiveness of the developed codes, maximum achievable theoretical limits for transition activity reduction are derived. **Chap**. 6 deals with coding schemes for throughput improvement. In this context, the advantages and drawbacks of state and transition coding are analyzed. Further, fundamental limits for power for performance are derived in the case of state coding. For transition coding, existing bounds for the maximum achievable data throughput rate are significantly improved. Afterwards, simple coding schemes are described and their effectiveness is analyzed and compared. An essential contribution of this work is that coding is combined with lower level interconnect optimization techniques like spacing and shielding. Finally, the problem of simultaneously decreasing power consumption and increasing performance is introduced and analyzed.

Part III An interconnect centric design flow that also integrates signal encoding for power and performance improvement is described in **Chap**. 7. In this context, a simultaneous placement and buffer planning algorithm is developed and combined with coding for throughput in order to prove the vast optimization possibilities in terms of performance and power available at higher levels of abstraction. Coding schemes can be first analyzed during the design and architectural specification step, where the necessary data properties can be extracted. The effectiveness of the codes can be evaluated and afterwards the schemes can be constructed during the synthesis phase. In order to further optimize the efficiency of the resulting codes, wire specific coding optimization can be performed during a refinement phase after the final impact of physical hierarchy and topology generation, as well as wire sizing, shaping, shielding, spacing, and splitting is exactly known.

Chapter 2

Technological Aspects of Interconnects and Effects of Input Patterns

Contents

Since the goal of the present thesis is to analyze and optimize power consumption and performance in on-chip interconnects at high levels of abstraction, the focus must be put first on identifying the lower level aspects and properties that have to be extracted in order to be shifted to the upper levels. Therefore, this chapter addresses the effects of technology scaling on interconnect structures. Moreover, the conjunction of the underlying physical effects and the transmitted signal characteristics is discussed.

Sec. 2.1 enumerates the essential characteristics of device and interconnect scaling. It is shown that the problems arising in on-chip interconnects have to be differentiated from

perspective of local, intermediate, and global wires. Further, **Sec**. 2.2 discusses several interconnect and driver models and analyzes their increasing intrinsic complexity. Finally, **Sec**. 2.3 discusses the effects of input signal patterns on crosstalk noise, delay, and power consumption in capacitively and inductively coupled interconnects, as well as the impact of process variations on interconnect parameters.

2.1 Technology Scaling and Interconnects

Technology scaling refers to the systematic rules employed to miniaturize devices while maintaining or improving their characteristics in terms of speed, power-efficiency and reliability. Those methodologies, together with advances in device integration and lithography, have resulted in a steady reduction of device feature sizes over the past years.

Technological issues form the underlying background for every analysis related to on-chip interconnects, to quote Carver Mead: *"Listen to the technology; find out what it's telling you."* In order to provide a solid foundation for an understanding of interconnect-related VDSM effects, this section discusses interconnect-related scaling and its influence on performance and power consumption. The goal is twofold: to show that both capacitive and inductive coupling will play an increasingly important role in nanometer high-speed IC design and that depending on the type of buses, i.e. local, intermediate (also called semi-global), or global, the two types of coupling will play different roles.

2.1.1 Device and Interconnect Scaling

Traditionally, the so-called constant-electric field scaling presented in **Tab**. 2.1 has been employed in order to realize smaller devices. The fundamental idea of this concept is to reduce the device geometries while maintaining a constant electric field. For this purpose, the supply voltage must be reduced by the same scaling factor, α, as the device geometry. Nonetheless, because of the non-scalability of the bandgap energy and the threshold voltage, and the exponential increase in leakage current appearing in VDSM technological nodes, the constant-electric field scaling seized to be effective. Therefore, the scaling factor used for shrinking the supply voltage had to be revisited by multiplying it with a correction factor ς in the so-called generalized scaling (see **Tab**. 2.1).

Nevertheless, the generalized scaling brings along two serious drawbacks. First, it increases the reliability issues related to the electric field increase inside the MOS devices. Secondly, the increase in power dissipation per unit area induces higher thermal requirements for the packaging. Consequently, one has to carefully choose the most adequate pair of supply and threshold voltages. The insulator thickness is reduced in order to increase the current drive and the supply voltage is scaled down in order to avoid breaking down the gate oxide. By decreasing the supply voltage, the dynamic component of the power consumption is also reduced but the driving strength of the gates is worsened.

Tab. 2.1: Constant field and generalized device scaling

	Constant Field Scaling	*Generalized Scaling*
Channel Length	$1/\alpha$	$1/\alpha$
Channel Width	$1/\alpha$	$1/\alpha$
Gate-Oxide Thickness	$1/\alpha$	$1/\alpha$
Electric Field	1	ς
Voltage	$1/\alpha$	ς/α
Doping	α	$\varsigma\alpha$
Gate Delay	$1/\alpha$	$1/\alpha$
Power Dissipation	$1/\alpha^2$	ς^2/α^2
Power Density	1	ς^2

This can be partially compensated by reducing the threshold voltage which at its turn implies an exponential increase in leakage power consumption.

At first glance, the theory of MOS device scaling suggests that signal wires should be scaled down by the same factor as active devices, so that the chip area can be reduced. However, due to increasing total number of devices integrated on a single and augmenting total die area, the number of required metal layers and the mean length of global and intermediate wires grow with advancing technological nodes. Moreover, as the wire cross section shrinks, the conducting characteristics of the wires degrade. In order to address these two conflicting requirements, interconnect scaling in modern sub-micrometer technologies can be split into two distinct components: *local* and *global* wire scaling. Note, that sometimes also a third component, i.e. the *intermediate* wire scaling, can be considered. Local wires refer to lower metalization layers, which are normally used to connect nearby gates within a given digital module. On the opposite, global wires are used for connections among blocks, and for power and clock routing. The fundamental differences between these two types of wires suggest different scaling strategies. Two typical scenarios [191] are depicted in **Tab**. 2.2.

A major concern for the scaling of local wires is to maintain the high integration density provided by the smaller device features. In order to accomplish this goal, both wire width and thickness are scaled down. Consequently, the wire cross-section is reduced, as well as its conductance per unit length. Since the capacitance per unit length is kept almost constant and the mean wire length decreases, the RC delay is not strongly altered by scaling. The main drawback of this approach is the increase in the current density of the wires, which reduces the reliability of the system. To cope with this problem, the wire thickness is reduced by a smaller factor in the so-called *quasi-ideal scaling*. Thus, the RC delay is improved by a factor of $\sqrt{\alpha}$, and the current density does not increase as rapidly as in the previous scenario.

Tab. 2.2: Wire scaling scenarios for local and global interconnect

	Local Wiring		*Global Wiring*	
	Ideal Scaling	*Quasi-ideal Scaling*	*Ideal Scaling*	*Constant Dimensions*
Wire Width	$1/\alpha$	$1/\alpha$	$1/\alpha$	1
Wire Thickness	$1/\alpha$	$1/\sqrt{\alpha}$	$1/\alpha$	1
Wire Length	$1/\alpha$	$1/\alpha$	$\sqrt{\alpha}$	$\sqrt{\alpha}$
Resistance	α^2	$\alpha^{3/2}$	α^2	1
Capacitance	1	≈ 1	1	1
RC Delay	1	$1/\sqrt{\alpha}$	α^3	α
Current Density	α	$\sqrt{\alpha}$	α	$1/\alpha$

Despite these important advantages, the technique suffers from a major drawback related to the increased aspect ratio of the wires. As the wire thickness gets larger than the width, the manufacturing process requires deep and narrow trenches which are difficult to produce. Furthermore, the capacitance between neighboring wires increases dramatically and becomes the dominant factor of the total wire capacitance. The consequence is higher crosstalk noise that degrades the signal integrity and modifies the power consumption of the bus line drivers [228]. Because local wires are generally very short, self and mutual inductance will not play an important role, and thus, local lines will be characterized almost exclusively by capacitive coupling. It is to be mentioned nonetheless, that continuously decreasing line resistivity and rise/fall times may eventually make the inductive behavior non-negligible even in local interconnects.

If the ideal or quasi-ideal scaling methodologies previously discussed were applied to global wires, an unacceptable performance loss would occur. The reason is the different mean wire length behavior for local and global wires. Since global wires connect blocks, their length depends on the total chip area. Actually, the mean length of global wires increases with a factor of approximately $\sqrt{\alpha}$. The consequence is an unacceptable increase in the RC delay and to palliate this problem, *constant dimension scaling* may be applied. In this case, the dimensions of upper layer wires are not modified, and, thus, an improved RC delay can be achieved (see **Tab**. 2.2). Obviously, the drawback is a drop in the routing resources at the upper levels.

2.1.2 Effects of Technology Scaling

Power consumption in digital circuits can be classified in two main components as a function of the dependency on the temporal variation of the input signals, namely the *static* and the *dynamic* power consumption [8, 28, 128, 138].

The dynamic power consumption refers to the the portion of the total energy dissipation related to the temporal signal variations. In digital CMOS circuits, the dynamic power is associated with two main phenomena, namely the capacitive switching current due to the charging and discharging of internal capacitances, and the short-circuit current that appears in the direct paths between supply voltage and ground created during switching [141, 142, 153]. During a low-to-high transition at the output of a CMOS gate, the PMOS pull-up network charges the output capacitance C_{out} by drawing from the power supply an energy equivalent to $C_{out}V_{dd}^2$, where V_{dd} represents the supply voltage. Half of this energy is dissipated in the resistive part of the circuit and the other half is stored in the capacitor. This stored energy is lost during an inverse switching. The mean value of the capacitive switching energy consumption per cycle is therefore:

$$\hat{E}_{switch} = \frac{1}{2} C_{out} V_{dd}^2 t_i, \tag{2.1}$$

where t_i denotes the probability of having a transition at the cell output per unit cycle, also called transition activity [52].

The exact value of the short-circuit energy dissipation depends on the duration of the direct path between the supply voltage and the ground. The energy consumed whenever the NMOS and PMOS nets are simultaneously on depends also on other parameters like cell load, internal transistor conductivities, and the input signal transition time. As shown in [142], the value of the short-circuit energy is given by:

$$\hat{E}_{short} = \frac{\beta}{12}(V_{dd} - V_{THn} - V_{THp})^3 t_r t_i, \tag{2.2}$$

where t_r denotes the rise or fall time of the input signals, β is a technological parameter, and V_{THn} and V_{THp} represent the threshold voltages of the NMOS and PMOS blocks, respectively. In well-designed digital CMOS integrated circuits, the short-circuit power consumption represents however only a small fraction of the total dynamic power, typically in the range up to 10%. To conclude, the total dynamic energy consumption depends on switching probability and operating frequency.

Traditionally, the static power consumption existing even in the absence of toggling in CMOS circuits due to leakage currents has been orders of magnitude smaller than the dynamic power consumption and therefore ignored. In recent VDSM technologies however, the so-called sub-threshold leakage increased so much that it cannot be neglected anymore [8,127,128]. The sub-threshold leakage current, I_{DS}, appears whenever the gate-source voltage, V_{GS} is below the threshold voltage. In the case of long-channel transistors, I_{DS} can be approximated by employing the equation of a bipolar transistor:

$$I_{DS} = k \cdot V_t^2 \cdot \mathrm{e}^{\frac{V_{GS}-V_{TH}}{nV_t}} \cdot (1 - \mathrm{e}^{-\frac{V_{DS}}{V_t}}), \tag{2.3}$$

where k and n are technology-dependent parameters, and $V_t = \frac{KT}{q}$ represents the thermal voltage. Thus, I_{DS} has a strong variation with temperature and threshold voltage. In short-channel devices, the drain voltage induces a decreasing of the threshold voltage,

an effect known as drain-induced barrier lowering (DIBL). DIBL has an important influence on the leakage current, as the latter increases exponentially with V_{DS} [52]. It is to be noticed that the sub-threshold current depends on input signals as those select a different conducting topology of the CMOS circuitry, and thus different drain-source voltages may appear [52]. With the continued scaling of the gate oxide to only a few nanometers, leakage currents due to tunneling are also increasing at fast pace [127]. Since the silicon dioxide layer thickness is reaching the limits of scaling, it has been proposed to employ high-K insulators [66]. Thus, a thicker gate layer can be used reducing thus the so-called gate-oxide leakage currents.

With increasing wire length, buffers are inserted in order to mitigate the effects of growing wire delay. Traditionally, the inserted buffers barely influenced the total area and power consumption of a system. However, repeaters are reported to become a problem at both chip- and block-level [165] as the percentage of total (local and global) repeaters in a design is projected to reach 35 % by the 45 nm technology node and even 70 % by the 32 nm node. This means that buffers will eventually be responsible for the majority of the die area and total (leakage-induced) static power consumption. The dynamic power consumption will be mainly influenced by the total interconnect structure capacitance.

Since clock frequencies augment, dynamic power consumption also increases, and due to the square dependency on the supply voltage it is very attractive to attempt reducing the supply voltage. As previously mentioned, this comes at the expense of higher leakage currents. In addition, the total leakage power consumption increases in interconnect structures due to the vast amount of required buffers. Moreover, with the steady growth of the interconnect structures, interconnect capacitance has become one of the main cause for dynamic on-chip energy consumption.

As seen in the previous subsection, the intrinsic gate delay is scaled down while the global interconnect delay is increased. Thus, the overall trend is that the interconnect-induced delay dominates the gate delay. As local interconnects are scaled down in such a way that the spacing is reduced while the aspect ratio increases, the coupling capacitances dominate the ground capacitances.

Moreover, with increasing interconnect length and signal rise times, inductive effects cannot be neglected anymore [191]. Therefore, inductive coupling will play an important role in global as well as in intermediate (also called semi-global) wires. However, an interesting observation can be added at this point. Being a long-range effect, inductive coupling allows neighbors of order higher than two to become inductive aggressors. In the case of neighbors of order one, one cannot know *a priori* whether they are inductive or capacitive aggressors. This is decided by wire geometries, propagation time, and rise times. Basically, we expect to have three major cases when inductive coupling is significant: first, the very inductive case when the first-order neighbor is an inductive aggressor; second, the medium inductive case for which the first-order neighbor is a capacitive aggressor but overall, the cumulated effect of all inductive aggressors dominates the effect of the capacitive ones; and third, the less inductive case when the capacitive effect of the

first-order neighbors outweighs the added effect of all inductive aggressors. As a first order approximation, we can expect the first and second case to appear more in global buses and the second and third to be typical for intermediate wires. In local lines, inductive effects will generally continue to be negligible without any loss in accuracy.

With steadily increasing importance of coupling capacitances and inductive effects in modern VDSM technologies, crosstalk between neighboring lines becomes a serious issue. Crosstalk has to be taken into consideration and accurately estimated in order to assure signal integrity and avoid malfunctions [20]. Other important noise-related problems are voltage fluctuations due to simultaneously switching gates, charge sharing and leakage currents [30]. Furthermore, the high density of flowing currents create hot spots, which at their turn may augment the temperature and the leakage current. In addition, the increasing current density through interconnects with shrinking geometries can cause electron migration. Therefore, reliability becomes an even more serious problem to be tackled in VDSM technological nodes [30].

Process variations are fluctuations in the value of process parameters. The impact of process and environmental variations on performance and power consumption has been increasing with each semiconductor technology generation [66, 183]. Variations in gate-length is considered to be the most critical device variation. They imply a shift in the DIBL coefficient, and thus a shift in V_{TH}. Moreover, channel doping variations increasingly influence random variations. The trends in the magnitude of process variations and their impact have been highlighted especially in [22, 66, 67, 183]. It is widely accepted that the increase in the variability of interconnect parameters such as wire width w, thickness t, height h, spacing s, resistivity ρ, surface roughness, and many others, are expected to increase significantly together with variability in gate-oxide thickness T_{ox}, power supply voltage, and threshold voltage. The impact of process variations has been shown to be important in the case of performance and momentous in the case of (leakage-induced) power consumption [183].

2.2 Interconnect Modeling

This section analyzes discusses several circuit models of increasing complexity that are employed for the analysis of the behavior of interconnect structures, i.e. the wires and their driving gates. The choice of one model or another is basically a trade-off between the required accuracy and the resulting computational overhead. On the one hand, simple RC models provide fast results but they are highly inexact for the analysis of modern high-speed interconnects. One the other hand, a full-wave model is extremely accurate by including all possible parasitics but because of its intrinsic complexity it can be applied only to relatively small structures.

2.2.1 PEEC Method and Distributed Models

In a wide sense, a *high-speed interconnect* can be defined as an interconnect which allows a signal with very fast slews to propagate in a very short time [1]. Fast propagation requires also very fast rise-times and when the rise time becomes comparable to the propagation time or the line losses are not negligible, the line actually isolates electrically the driver from the receiver. Within the transition time, the interconnect acts as the load to the driver and as the input impedance to the receiver. Thus, various transmission line effects, such as reflection, crosstalk, and overshoots have to be taken into consideration [1,30,117,156]. Depending on their structure, signal rise time, and operating frequency, interconnects can be modeled as lumped, distributed, or full-wave models.

The most important criterion employed for classifying an interconnect is based on its *electrical length*. A wire is considered to be electrically short if, at the highest operating frequency of interest, the interconnect is much shorter than the corresponding wavelength [1]. In general, the highest operating frequency is determined by the rise and fall times of the propagated signal. Even though the frequency spectrum of a trapezoidal pulse is infinite, the energy of the signal is concentrated in the lower part of the spectrum and rapidly decreases with increasing frequencies. Hence, the signal spectrum can be considered finite without affecting the signal waveform. For this purpose, the concept of *significant frequency*, f_s, has been proposed to reduce the complexity of the required information. For a trapezoidal pulse, f_s is defined as $0.34/t_r$, where t_r represents the pulse ramp time (rise time). Less than 15 % of the spectral components are at higher frequencies than f_s, and their overall magnitude is very small [30]. In some cases, the more conservative limit of $1/t_r$ may be used [1].

At low frequencies, wires are electrically short, and electromagnetic phenomena descriptions can be reduced to electric models. Thus, interconnects can be accurately modeled with lumped RC or RLC circuit models. Nevertheless, due to faster rise times and increasing interconnect lengths, the electrical length of interconnects becomes a significant fraction of the operating wavelength, and transmission line effects must be taken into account. Important effects like resistive shielding cannot be ignored anymore and lumped models become inadequate because they cannot accurately predict crosstalk, rise time, or delay [30, 117]. As a result, transmission line models based on the transverse electromagnetic mode (TEM) assumption are required. Moreover, when dimensions are electrically large, the structure can be broken into a set of electrically small substructures. Each of these substructures is equivalent to a lumped model based on the so-called *per-unit length (PUL) parameters*. The PUL parameters for inductance, capacitance and conductance are governed by the fields external to the conductor and are determined as a static solution to the Laplace equation in the transversal plane of the line. In contrast, the entries in the PUL resistance matrix are governed by the interior fields [117, 137]. Consequently, the PUL parameters contain the entire cross-sectional structural dimensions of the interconnect.

The main assumption made to derive the so-called *telegrapher's equation* [117] is that no field components exist in the direction of propagation, which is the case when there are losses neither in the conductors nor in the dielectric material. Nevertheless, both conductors and dielectrics are imperfect and small losses captured by the PUL resistance parameter do exist. The medium may be lossy and not violate the TEM assumption as long as it is homogeneous [137]. Lossy conductors invalidate the TEM field structure assumption, but if conductors are characterized by "small" losses, it is still possible to find an approximate interpretation in terms of quasi-static voltage and current in the orthogonal plane, and therefore, an electrical model [117]. This assumption is referred to as *quasi-TEM*. In practical situations, interconnects may need to be modeled as nonuniform lines, and in this case, the PUL parameters are functions of the distance [1]. Details about different analytical and numerical methods for determining the PUL parameters can be found in [117, 137, 255].

From the theoretical point of view, self and mutual inductances are loop-dependent quantities and can therefore be determined only if the whole current loop, i.e. the return path, is known [7, 51, 70, 115, 114, 155]. However, the return path is especially difficult to determine not only because of its frequency dependency, but also because of the lack of a ground plane in higher metal layers. In order to cope with this issue, Ruehli proposed in [155] an alternative inductance extraction approach based on partial element equivalent circuits (PEEC), which is well-suited for circuit simulation as it depends only on circuit geometry. The fundamental idea of PEEC is that the partial inductance of a wire is considered the inductance of that wire as it forms a loop with infinity. Thus, there is no *a priori* knowledge required and no need to specify any return path, as the smallest current loops for high frequencies and the least-impedance return paths are determined by simulation, e.g. SPICE.

As indicated by He in [58], three foundations for inductance extraction emerged as a result of the PEEC method. First, the partial self inductance of a wire depends solely on the wire geometry itself, that is length, width, and thickness; secondly, the partial mutual inductance of two wires is solely decided by the geometry of the two wires themselves (spacing, lengths, widths, thicknesses); and last, the mutual inductance between any two orthogonal wires is negligible.

In the PEEC approach, the interconnects are subdivided into small surface and volume elements. Partial inductances and capacitances are computed from these elements and the resulting circuit elements are combined with each other into a complete PEEC circuit. For interconnect structure sizes which are much smaller than the smallest wavelength of interest, one can assume that the field instantly travels through space from one point to another. The assumption of quasi-stationarity allows a description of time dependent fields from static field calculations. Thus, PEEC models are RLC circuits where individual elements are extracted from the geometry using a quasi-static (non-retarded) solution of Maxwell's equations [1]. However, at very high frequencies the PEEC model is inaccurate as it does not consider the effect of the finite speed of light, i.e. retardation. Therefore, the

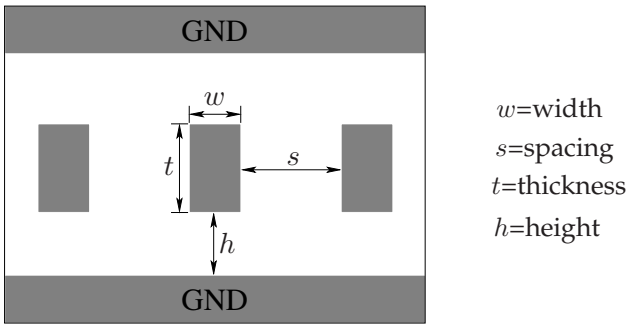


Fig. 2.1: Geometry parameters of a VDSM bus

PEEC approach has been extended to geometries where the size of the critical coupling distances is no longer short compared to the wavelength. The retarded PEEC (rPEEC) models include the retardation effect and provide a full-wave solution [156, 157].

In order to efficiently and accurately determine the signal characteristics of a bus, it is of utmost importance to choose the most appropriate model for on-chip interconnects. Even though a full-wave model would always give the correct solution, such approaches are computationally extremely intensive. There are several works in the literature proposing and dealing with qualitative and quantitative methods for choosing the most appropriate interconnect model, i.e. an accurate model of least complexity [6, 26, 30, 37, 39, 69, 70, 109]. In the following, some of the most used methods and figures of merit are briefly presented.

When the line capacitance becomes comparable to the load capacitance, signal delay propagation cannot be neglected anymore for a correct delay analysis. Additionally, in the case of longer lines, signal propagation is worsened due to the increasing line resistance, and therefore RC models have to be employed. However, simple lumped RC models can only be efficiently used if the interconnect induced propagation delay is considerably smaller than the rise time of the signal propagating through the interconnect. Thus, the distributed nature of the line impedance would not be modeled and the line has to be split into several cascaded lumped segments. The propagation delay induced by each of those segments, $t_{p,seg}$, must be much shorter than the rise time [1, 41, 133]. A typical acceptable delay through a segment is:

$$t_{p,seg} \lesssim \frac{t_r}{10}. \tag{2.4}$$

In VDSM technologies, the aspect ratio of bus wires increases mainly because of two factors. On the one hand, in order to preserve a high integration, wires have to be scaled down together with devices. On the other hand, as previously shown, the wire characteristics are reduced at a faster pace, and in order to keep their resistivity at acceptable levels, the thickness must be increased. The result are densely packed (s decreases) wires with higher aspect ratios (t becomes higher than w) (see **Fig**. 2.1). Thus, the coupling capacitances between neighboring wires steadily increase with every new technology node

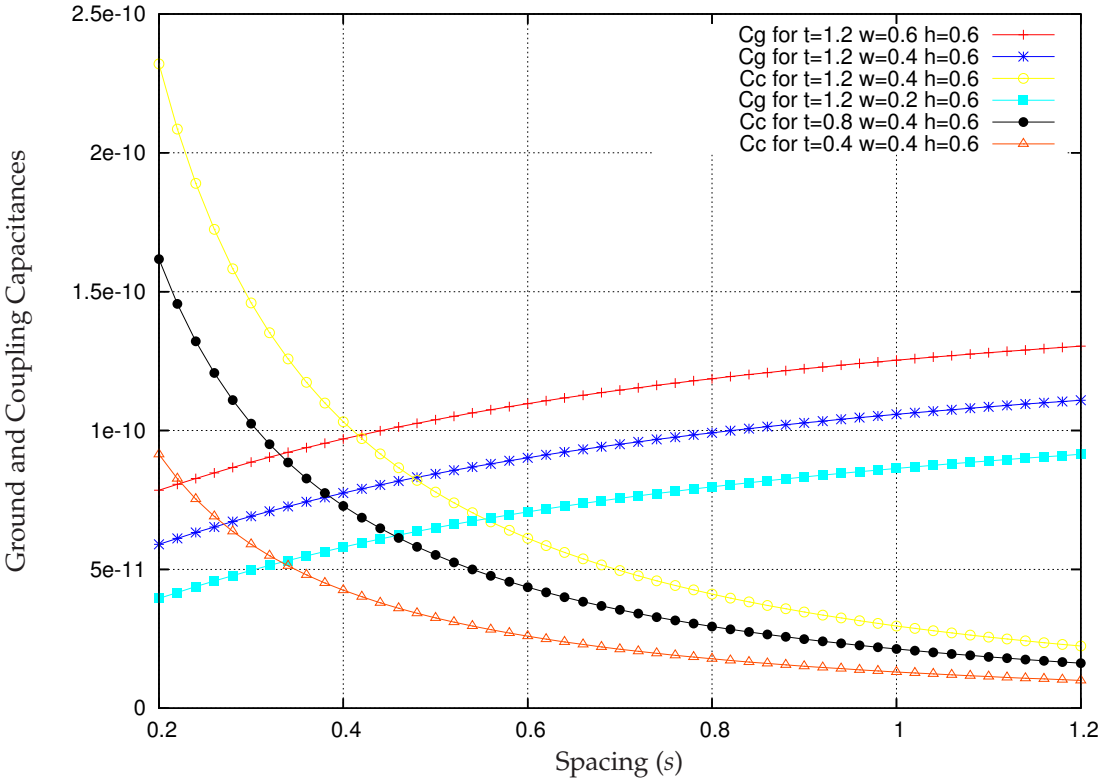


Fig. 2.2: Ground and coupling capacitances in VDSM technologies

and might even dominate the ground capacitances that appears between each line and the upper and lower metal layers [67].

Wong et al. developed in [200, 201] empirical formulas for the coupling (C_c) and wire or ground (C_g) capacitances. For equal distance to the upper and lower metal layer, those formulas become:

$$C_g = \varepsilon_0 \varepsilon_r \left[\frac{2w}{h} + 2.04 \cdot \left(\frac{s}{s + 0.54h} \right)^{1.77} \cdot \left(\frac{t}{t + 4.53h} \right)^{0.07} \right], \tag{2.5}$$

$$C_c = \varepsilon_0 \varepsilon_r \left[1.41 \cdot \frac{t}{s} \cdot \mathrm{e}^{-\frac{4s}{s+8.01h}} + 2.37 \cdot \left(\frac{w}{w + 0.31s} \right)^{0.26} \cdot \left(\frac{h}{h + 8.96s} \right)^{0.76} \cdot \mathrm{e}^{-\frac{2s}{s+6h}} \right], \tag{2.6}$$

where $\varepsilon_0 = 8.8541878176$ pF/m and ε_r are the electrical permittivity in vacuum and the relative dielectric constant, respectively. Similar formulas have been developed among others by Eo and Eisenstadt in [43, 44, 73] and by Sakurai in [160].

Fig. 2.2 shows the dependency of the ground and coupling capacitance on the line spacing. It can be observed that with decreasing s, the ground capacitance is slightly reduced due to the fact that the a higher part of the fringing capacitance contributes to the coupling capacitance. Further, the coupling capacitance rapidly increases with decreasing s. Typical values have been chosen for sub-100 nm technologies as given in [125].

As mentioned in the previous section, global lines are generally wide and exhibit low resistance. In addition, the increase in clock frequencies drives the rise and fall

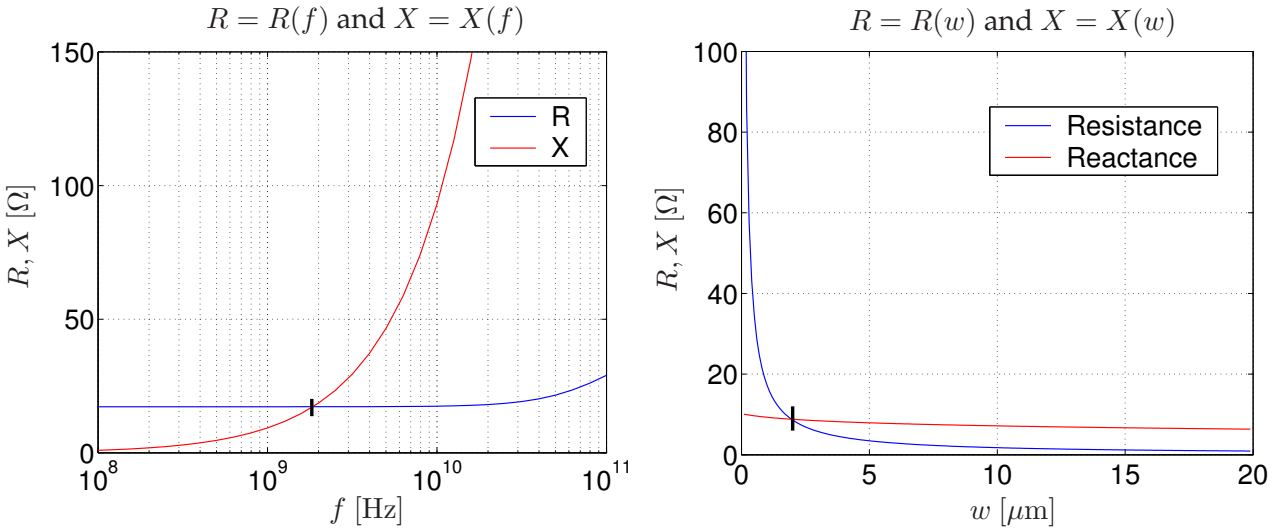


Fig. 2.3: Reactance and resistance of a single wire versus frequency for nwinc = 25, nhinc = 25 ($w = 1$ µm, $l = 1000$ µ)

times to steadily decreasing values and thus, the significant frequency is pushed into the GHz domain. Therefore, the line inductance affects even more the timing characteristics of on-chip interconnects and has to be included into a precise model. As a simple rule of thumb, inductive effects are to be considered whenever the inductive impedance is comparable to the resistive impedance. This issue is illustrated in **Fig**. 2.3 which shows that at high frequencies the inductive impedance can dominate the resistive impedance even though the latter also increases due to the skin and proximity effects[1]. The impedances have been extracted with FastImp [75] with the parameters as indicated in **Fig**. 2.3 (see [255]).

Resistance shows a higher sensitivity on conductor width than inductance, and because inductance decreases very slowly with increasing width, tuning the width is not a very effective technique for minimizing the inductance. Moreover, **Fig**. 2.4 shows the effect of geometry on reactance or, equivalently, inductance, because a single frequency is chosen. The most important conclusion to be drawn from **Fig**. 2.4 is that on-chip inductance does not scale linearly with conductor length. Actually, the inductance starts to deviate strongly from the linear approximation shown in the same figure especially towards smaller values of l. It is to be noticed that $X = \omega L$, so that X is proportional with L for fixed values of f. The width has been varied from 0.1 µm through 20 µm in steps of 0.1 µm, while the length has been varied from 1 µm through 10,000 µm in steps of 50 µm. The right figure compares results from FastHenry extractions, **Eq**. (2.7), and linear approximation of the simulated curve. The analytical and numerical results present a very good agreement.

[1] As explained in [30], wire splitting can be employed for limiting the skin and proximity effects, and thus the associated increase in line resistivity

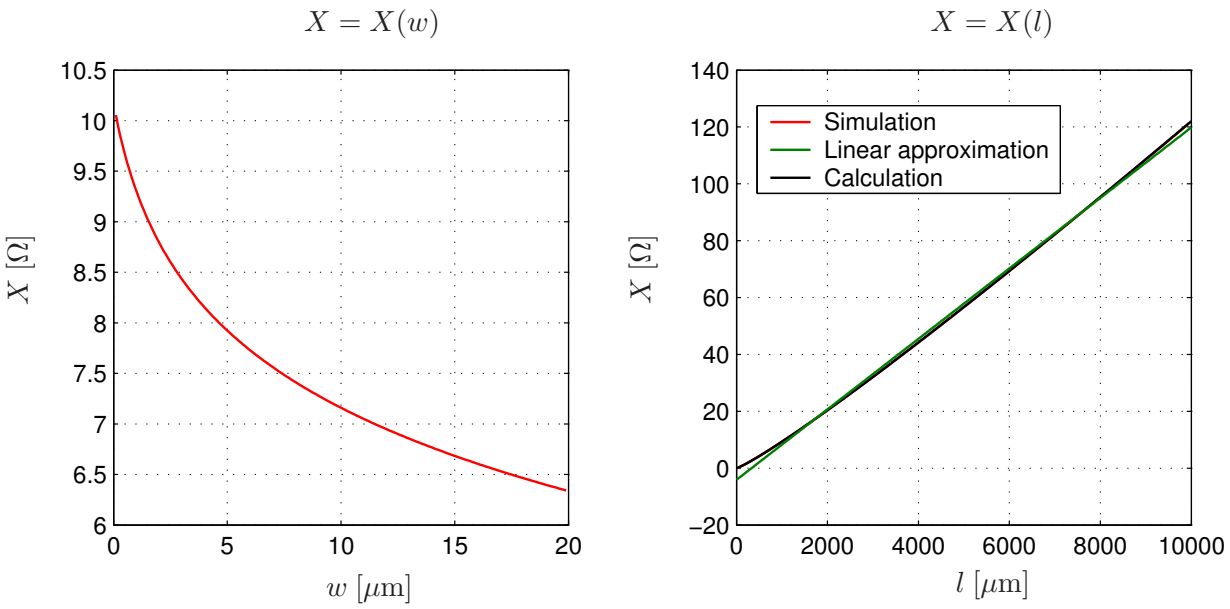


Fig. 2.4: Reactance of a single wire at 1 GHz depending on wire width, w, and length, l, for $w = 1$ µm (right figure), $l = 1000$ µm (left figure), nwinc = 11, and nhinc = 11

The scalability issue of inductance with length can be also shown by the following simple formula for partial self-inductance [57,150]:

$$L = \frac{\mu_0 \mu_r l}{2\pi} \left[\ln\left(\frac{2l}{w+t}\right) + 0.5 - k \right], \tag{2.7}$$

where $k = f(w,t)$, $0 < k < 0.0025$, is defined by means of the geometric mean distance as shown in [169]. **Fig**. 2.4 shows the agreement of **Eq**. (2.7) with FastHenry extraction. The argument of the logarithm and k are the cause for the super-linearity of inductance with respect to wire length l. One of the shortcomings of **Eq**. (2.7) is that it neglects the skin effect and thus, it may lead to inductance overestimation at higher frequencies. A further drawback of **Eq**. (2.7) is the neglection of the internal inductance.

A similar formula exists for the mutual inductance, M, between two parallel and identical wires [57,58,117,133,151,152]. The formula shows that mutual inductance is a super-linear function of the length too and is given by:

$$M = \frac{\mu_0 \mu_r l}{2\pi} \left[\ln\left(\frac{2l}{p}\right) - 1 + \frac{p}{l} \right] = \frac{\mu_0 \mu_r l}{2\pi} \left[\ln\left(\frac{2l}{w+s}\right) - 1 + \frac{w+s}{l} \right], \tag{2.8}$$

where p is the inter-wire pitch, while $\mu_0 = 4\pi \cdot 10^7$ N/A^2 and μ_r represent the free-space (vacuum) magnetic permeability and the relative permeability, respectively. Thus, mutual inductance is a super-linear function of the length too. Nevertheless, as no good approximation formula exists for the mutual inductance between two parallel lines of unequal length, it is necessary to use accurate parasitics extracting tools or field solvers – such as FastHenry, Synopsys' Raphael and Star-RCXT, Cadence's Assura, Sequence's Columbus-

RF, OEA's Cheetah and Metal [30, 55], to name only a few – in order to determine the inductive coupling more accurately than with closed-form equations [100].

In order to select the most appropriate model with regard to inductive behavior, a selection process like the one described in [117] must be performed. First, the rise time of the signal after the driver has to be compared with the propagation time, t_p. Therefore, the input rise time is not necessarily a well-suited parameter for selecting the line model. Nevertheless, when the input rise time is much greater than the propagation time, non-inductive models can be used to simulate and check the value of the rise time at the driver output. The inductive model is not to be used if this approximate rise time is still greater than the propagation time. Note that the rise time is often compared to $2t_p$ in order to consider reflection [117].

Secondly, in order to assess the possibility of ringing and thus the need of an inductive model, one ought to compare the driver impedance, Z_d, with the characteristic impedance of the line, Z_l. The output impedance of a line, Z_o, is usually capacitive and thus very small compared to the line impedance, $Z_o < Z_l$. Consequently, the condition for ringing is $Z_d < Z_o$, and, thus, lower impedance drivers are candidates which may require an inductive interconnect model. As shown in [117], if the driver and load have linear characteristics, the conditions for the appearance of ringing can be summarized in general as follows:

$$Z_d \leq Z_o \leq Z_l, \tag{2.9}$$

$$Z_d \geq Z_o \geq Z_l. \tag{2.10}$$

Apart from the ratio between signal rise time and time-of-flight and the relation between driver and line impedances, there is another important factor in correctly determining the interconnect model to be employed: the damping of the interconnect line [69]. The damping factor of an RLC line is given by:

$$\xi = \frac{\tau_{RC}}{2\tau_{LC}} = \frac{R_t C_t}{2\sqrt{C_t L_t}} = \frac{Rl}{2}\sqrt{\frac{C}{L}}\,,$$

where R, L, and C are the resistance, inductance, and capacitance per unit length respectively, l is the length of the line, R_t, L_t, and C_t are the total resistance, inductance, and capacitance of the line, respectively, and τ_{RC} and τ_{LC} are the RC and LC time constants of the line, respectively. When ξ decreases, which actually means that the effects due to reflections increase, the RC model becomes inaccurate.

In [69], the following figures of merit, based on transmission line analysis, have been proposed. A digital signal that is propagating in an underdriven uniform lossy transmission line exhibits a significant inductive behavior if the line length l satisfies the following condition:

$$\frac{t_r}{2\sqrt{LC}} < l < \frac{2}{R}\sqrt{\frac{L}{C}}. \tag{2.11}$$

This range depends upon the parasitic PUL impedances of the interconnect and the rise time of the signal. This range may be nonexistent in certain cases, namely if:

$$t_r > 4\frac{L}{R}. \tag{2.12}$$

When this condition holds, inductance is not important for any interconnect length [69]. **Ineq**. (2.11) represents basically the conjunction of two rules that have an intuitive circuit interpretation [116]. The rule on the left side has been introduced to ensure the waveform agreement between the analytical solutions of the characteristic impedance of the transmission line and its RC approximation [30, 69]. It is to be mentioned, that the velocity of the electromagnetic signal propagation along a line is $v_c = \frac{1}{\sqrt{LC}}$. Thus, the term on the left side, $2\sqrt{LC}l = 2\sqrt{L_tC_t}$, is equal to twice the time required by the electromagnetic wave to travel from one end of the line to the other, that is the round trip time-of-flight [30,116]. Alternatively stated, the line length l should be a significant fraction of the shortest wavelength of significant signal frequencies. The left inequality restricts significant inductive behavior to electrically long interconnects.

The rule on the right side makes sure that the equivalent RLC circuit is underdamped ($\xi < 1$). As shown in **Fig**. 2.5, given a line with specific PUL parameters, the inductive behavior is confined to a certain range of interconnect length. The lower bound of this range is determined by the electrical size of the interconnect, while the upper one is given by the dampening factor of the line [116]. It is to be mentioned, that several authors employ modified versions of the rule on the right side. For instance, Ho et al. found in [61] that a total line resistance greater than approximately $2.5Z_0$ would attenuate the possible ringing effects to negligible levels while Moll and Roca compare R_t with $3Z_0$.

Alternatively, the double inequality 2.11 can be interpreted as a bound on the total line inductance L_t. As indicated in [116], the interconnect exhibits non-negligible inductive characteristics if the following two conditions hold:

$$L_t > \frac{1}{4}\frac{t_r^2}{C_t}, \quad \text{and} \tag{2.13}$$

$$L_t > \frac{1}{4}R_t^2C_t. \tag{2.14}$$

The region for total interconnect inductance with non-negligible inductive behavior is given in the right side of **Fig**. 2.5.

The process for the selection of an inductive or non-inductive line model can be found in form of an algorithmic scheme in [117], while a more detailed method is presented in [30].Briefly, we can also say that if the signal propagation delay is much less than the rise time and long lines are too lossy, inductance need not be taken into consideration. In the case of short lines, the propagation time is too small compared to the transition time, and, in general, inductance is not important for any length if the effect of attenuation comes into play before the effect due to the rise time vanishes [69]. Nevertheless, the aforementioned rules are rather loose, and when mutual inductance plays an important role, finding such figures of merits is a very tedious task and still under research [30].

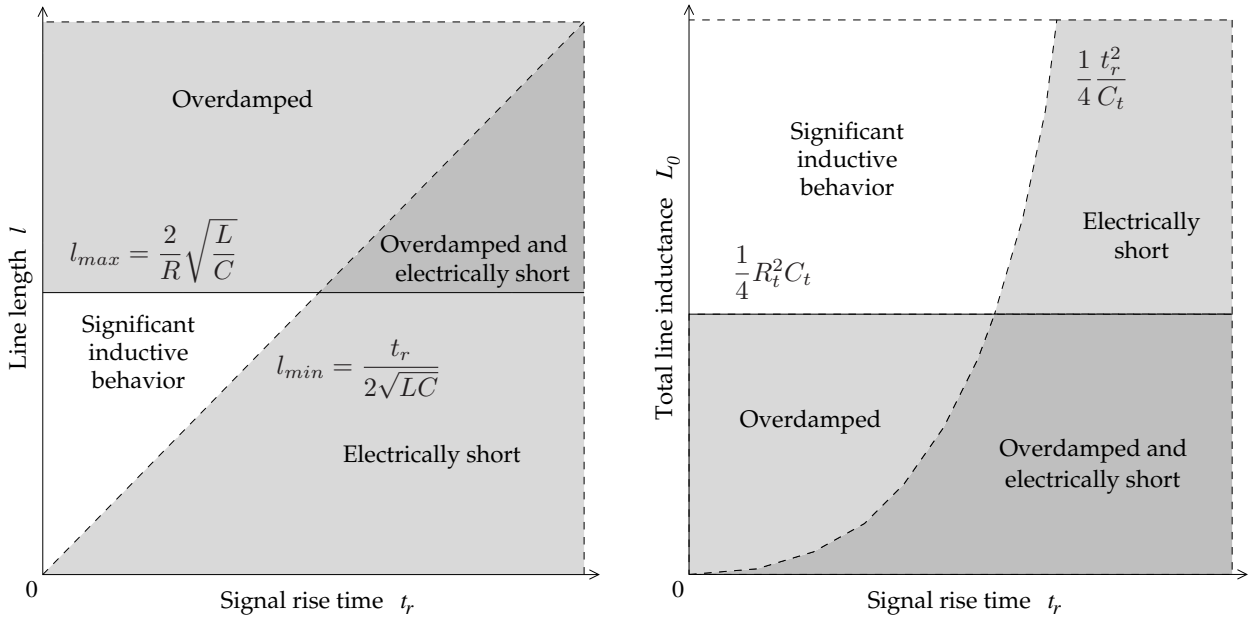


Fig. 2.5: Figure of merits for inductive behavior (after [116]). The ranges of interconnect length and total line inductance where the signal propagation exhibits significant inductive behavior are bounded by the conditions of large electrical size (dashed line) and low dampening (solid line)

To conclude, each interconnect line exhibits at high frequencies not only an associated self-inductance but also a corresponding mutual inductance to all near and far neighboring lines. Thus, interconnect models evolved from being ignored, to the trivial lumped capacitive model, to the widely used distributed RC model (resistance - capacitance) and later to the distributed RLC model (RC with self-inductance). Nowadays, the mutual inductances are also included in the so-called full RLC or $RLMC$ models (RLC with mutual inductance). In the following section, the abovementioned distributed models are employed in order to compare and analyze the effects of capacitive and inductive coupling on crosstalk, signal delay, and power consumption.

2.2.2 Driver Modeling and Gate Characterization

Classically, the analysis of the total delay induced by a gate driving an interconnect network has been addressed by splitting the problem into two simpler ones: separate computation of buffer delay and intrinsic delay through the wire (also referred to as time-of-flight). The major benefit of such an approach is that it isolates the nonlinearities from the linearities, because the parasitics that are associated with MOS transistors show significant nonlinearities and the wire parasitics are linear [164]. In order to determine the equivalent delay of a buffer, the complete network is abstracted as an equivalent load. The delay is then a function of the input transition time and that equivalent load. Key elements of this methodology are the estimation of the equivalent (or effective) load, which must not always be just a capacitance, and the delay of the wire. In order to shorten de-

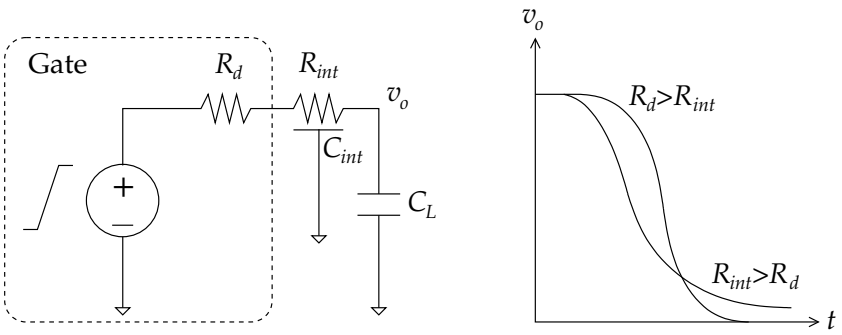


Fig. 2.6: Resistive shielding effect

sign cycles and reduce costs, gate and cell delays are precharacterized for static timing analysis. In general, gate and cell delays as well as slews are expressed in an empirical fashion as functions of load and input slew as explained in [26, 164].

Basically, there are two widely accepted techniques to gate delay modeling. On the one hand, empirical expressions or look-up tables for delay and output transition as a function of input transition time and load capacitance can be derived (the so-called k-factor equations). On the other hand, a switch-resistor model consisting of an empirically fitted linear resistor can be employed [26, 199]. The main benefit of the second method is the inherent modeling of the coupling with an RC interconnect. However, the drawback comes from the fact that only a single resistor is used for capturing the gate behavior. In order to cope with this issue, more complex models consisting of time and slew dependent non-linear resistances have been proposed [16, 173].

When the gate load is purely capacitive, the gate delay and output signal can be completely precharacterized as functions of the signal rise time, t_r, and the load capacitance, C_L. Nonetheless, because of the combination of the scaling-induced line resistance increase and the gate output resistance reduction, the so-called *resistive shielding* effect appears. If the gate resistance is greater than the line resistance, i.e. $R_d > R_{int}$, then the gate delay is accurately defined by means of the total load capacitance. However, as shown in **Fig**. 2.6, when the line resistance is greater than or comparable to the driving resistance, the gate does not "see" all the capacitance (the high interconnect resistance shields the capacitance, hence resistive shielding) and the output voltage, v_o, switches faster. Due to the increased line resistance, the total delay is eventually increasing [26]. The resistive shielding can be overcome quite simply by employing the so-called *effective capacitance* concept, that is an equivalent capacitance which accurately models the delay with resistive shielding [36, 76, 130, 140]. For more accurate analyses, one can introduce more effective capacitances or even waveform-dependent capacitances [38].

In the case of interconnects that can be characterized with equivalent capacitances that correspond to the effective capacitances employed for gate precharacterization, the aforementioned two-step delay approximation can be used. Thus, the gate output signal is approximated with a saturated ramp and afterwards, the interconnect delay is determined

by applying the resulting ramp as input waveform [26]. Nonetheless, with growing line resistance the saturated ramp approximation gets inaccurate. Therefore, in the Thévenin gate model, the gate is replaced by a constant resistor R_d and a time-varying voltage as shown in **Fig**. 2.6. As explained in [26,164], the shape of the waveform is determined in an iterative manner as a function of the effective capacitance. Generally, the Thévenin voltage is modeled by a saturated ramp characterized by a transition time, Δt, and a delay t_0 [26,164].

In the previous subsection, it has been shown that in VDSM technologies, high-speed interconnects as gate loads cannot be modeled any longer by purely resistive-capacitive loads [26, 164]. When inductive effects appear, the Thévenin and effective capacitance methodology might become inaccurate. Nevertheless, the technique can be enhanced to cope with such effects. For instance, piecewise linear Thévenin voltage source models [4] or multi-ramp driver models with RLC interconnect loads [194] have been proposed. Other important efforts have been put into characterizing the intrinsic delay of a buffer when loading lines are dominated by inductive effects [2,26,77,164]. It has been observed that line inductance can partly shield or hide the far-end load modifying thus the effective capacitance. Nonetheless, for common inductive cases, i.e. low to medium inductive coupling, the Thévenin method is accurate enough for delay analysis, and for this reason it is employed in the current work.

2.3 Effects of Input Patterns on Crosstalk, Delay, and Power

The goal of this section is to analyze and show the dependency of crosstalk, signal delay, and power consumption on the input patterns. For this purpose, typical scenarios for global and intermediate buses are chosen. Furthermore, the utmost importance of choosing the appropriate model is highlighted since models that are less complex but not accurate enough can finally lead to results that are far from reality.

2.3.1 Simulation Environment

Fig. 2.7 shows a typical VDSM interconnect structure. Eight signal lines are closely spaced and sandwiched between a V_{dd} and a ground (GND) line on top of an orthogonal layer. As described in **Fig**. 2.8, such parallel wires can be modeled by being partitioned into the least number of segments that provide sufficient accuracy [43,78,79]. For simplicity, only four lines are illustrated. In the absence of a closely situated dedicated ground plane, the capacitances that model the neighboring orthogonal layers cannot be directly connected to the ground as the current flowing into the orthogonal wires has to travel a long way to the drivers or receivers and afterwards, through those devices to the ground node. Thus, in some cases, a single ground node can bee too optimistic. Nonetheless, as explained in detail in [30], the orthogonal layer can be eliminated if treated as a so-called supernode.

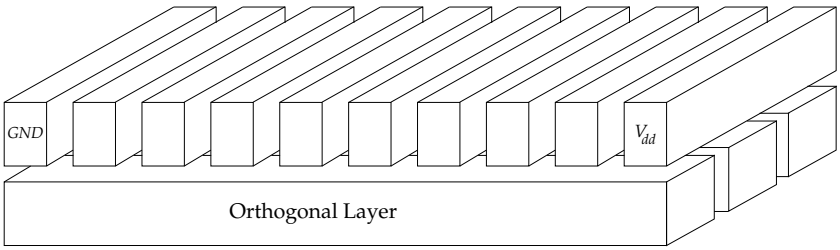


Fig. 2.7: Geometry of a bus on top of an orthogonal layer

In this case, the capacitances to the upper and lower metal layers are distributed on the neighboring lines.

In order to reduce inductive coupling, dedicated ground planes can be inserted between the metal layers. Those ground planes do not only have the property of isolating the metal layers, but also provide closely-spaced and low-impedance return paths. Thus, the orthogonal layer does not have to be eliminated and the modeling process is slightly less complex. It is important to notice, that the same effects appear in the case of both models. The only difference is however, that in the case of the latter model, inductive effects appear at higher values of the rise time and/or wire length. In order to keep the simulation complexity at manageable sizes, the second model is employed throughout this section. Moreover, skin and proximity effects proved to be negligible for the chosen set of interconnect parameters and have not therefore been included in the models [255].

In **Fig**. 2.8, a 4-segment $RLMC$ model for a 4-bit wide interconnect topology (plus one ground line) is shown. For illustration purposes, device symbols are given only for some segments and mutual inductances are only indicated for the third segments of lines 1 and 3 and for the fourth segment of lines 1 and 2. It can be observed that when ignoring the quiet power grid lines, we require for the full $RLMC$ model $N \cdot S$ resistances, $N \cdot S$ self-capacitances, $(N-1) \cdot S$ mutual capacitances, $N \cdot S$ self-inductances, and $N \cdot S(N \cdot S-1)/2$ mutual inductances, or more precisely, inductive coupling coefficients, where N denotes the number of conductors and S the number of segments. The number of segments is increased with increasing wire length to maintain sufficient accuracy. In order to reduce the order of the $RLMC$ model, several methods for making the inductance matrix sparse have been proposed [50,30,26]. Nevertheless, in order to avoid stability problems, one has to make sure that the sparsified matrix is positive definite. Since the focus is to employ a precise but not necessarily fast model, Model Order Reduction (MOR) techniques and automatic generation of reduced accurate circuit models for interconnects [74] are beyond the scope of this thesis. Consequently, the full $RLMC$ model is employed. Notice that when dedicated ground planes are used in order to reduce inductive effects, the return paths are much shorter. Furthermore, if the resistivities that result in the ground plane are small enough, then a single common ground can be used for the whole bus model [30].

In this section, a 500 µm and a 1000 µm long 5-bit wide buses are modeled. The buses are considered to be placed above a dedicated ground plane. Every line has been split

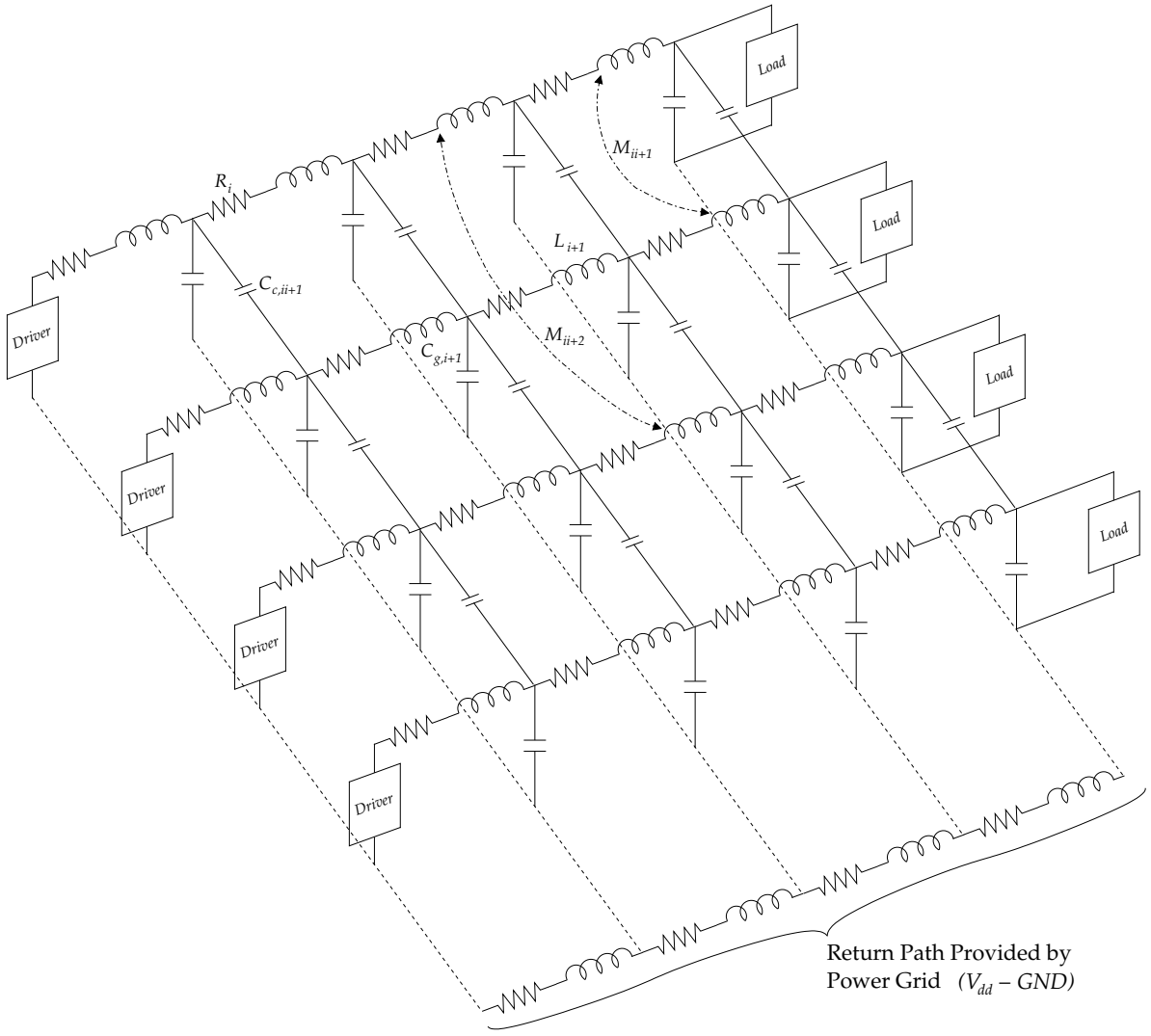


Fig. 2.8: Distributed full $RLMC$ interconnect model with power grid lines as return paths (after [78,79]): line resistance R_i, ground capacitance $C_{g,i}$, coupling capacitance $C_{c,i}$, self-inductance L_i, and mutual inductance M_{ij}, where i and j are the line indexes.

into 10 segments. The thickness, width, and spacing of the lines as well as the distance to the lower and upper metal layers are all considered to be 1 µm. The values have been chosen according to the upper layers characteristics of a generic 100 nm 1.8 V technology [125]. They represent typical scenarios for parallel buses in global and intermediate parallel buses, as there can appear, depending on the rise times, significant capacitive and inductive coupling effects. Note that, the employed transistor models used to implement the buffers also correspond to such a technological node [40].

Fig. 2.9 illustrates the parameter extraction flow. FastHenry [75] was used to extract the total resistances and inductances. In addition to material constants for metal (copper) and dielectric (SiO_2) and discretization values for the interconnect structure under consideration, the input data is specified by the number of wires N, wire length l, wire width w,

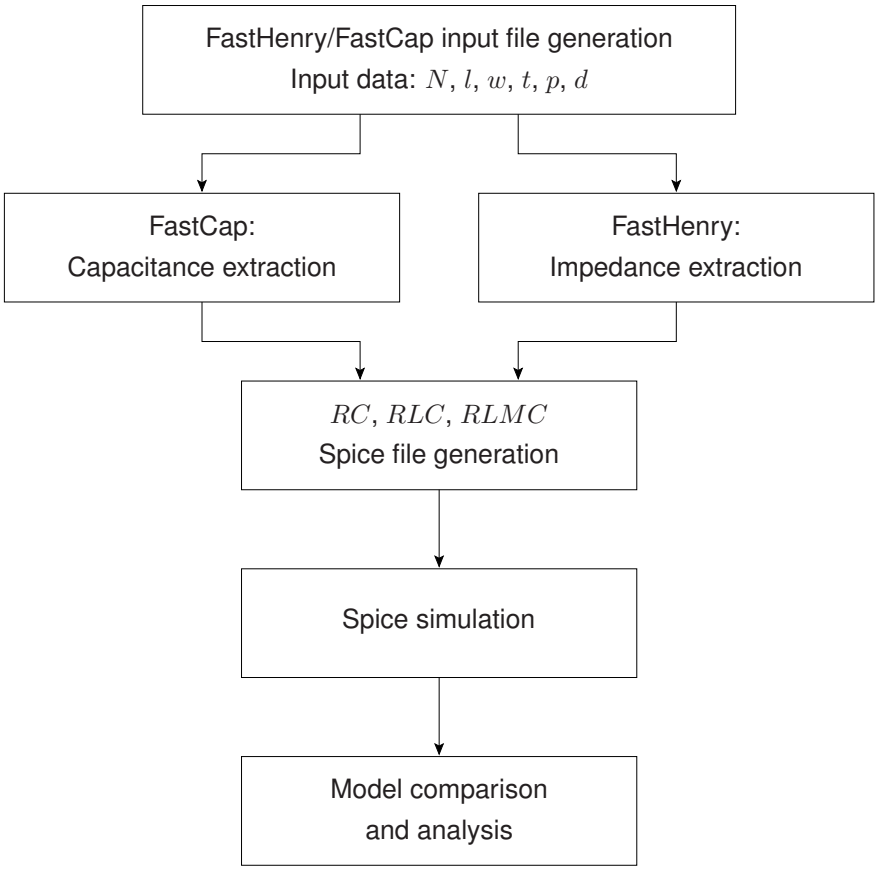


Fig. 2.9: Parameter extraction flow

wire thickness t, wire pitch p, and distance between two metal layers d. In order to obtain the required distributed parameter values, the input file for FastHenry has been adapted by splitting the five wires ($N = 5$), each of 1 mm length, into 10 segments ($S = 10$) closely spaced to one another. Thus, 50 resistances, 50 line inductances, and 1225 mutual coupling coefficients have been extracted and written into a SPICE netlist. Furthermore, the total ground and coupling capacitances have been computed with FastCap [123]. A total of 50 self-capacitances and 40 mutual capacitances have been extracted and afterwards used to complete the aforementioned SPICE netlist.

An approximate range for the driver impedance is of 50 to 300 Ω, as indicated in [117]. The lines in the simulations are driven by inverters with 75 Ω output resistance. It is to be mentioned that with every new VDSM technology node, drivers with decreasing impedance can be employed, and in the case of low-impedance drivers, there may be a greater need to employ inductive interconnect models.

At frequencies corresponding to rise times in the order of picoseconds, there may be significant skin depth to be considered [30]. Therefore, FastHenry simulations have been performed, which show the error in extracted interconnect parameters as a function of frequency at increasing values for volume discretization parameters. From these

results, it could be seen that for frequencies leading to rise times larger than approximately 25 ps, the influence of the skin effect on extracted interconnect parameters can be neglected. Apart from geometry and material constants, this value for the rise times also depends on the desired accuracy of extracted parameters. We have therefore restricted our simulations to rise times larger than 25 ps and present results for 25 ps, 50 ps, and 75 ps, which correspond to the significant frequencies of 13.6 GHz, 6.8 GHz, and 4.53 GHz respectively. The highest significant frequency, 13.6 GHz, has a corresponding wavelength of $\lambda = 22.059$ mm, which is much larger than the maximum simulated interconnect length of 1 mm. Thus, based on the qualitative and quantitative measures given in **Sec**. 2.2, quasi-stationarity can be assumed without any loss of accuracy and thus, the non-retarded PEEC method can be employed.

The error involved with various discretizations of conductors into segments was observed in further FastHenry simulations to find the optimal trade-off between accuracy and computation time. From these simulations the number of segments has been found that is necessary for the extracted impedance parameters to converge to their final values within a specified range. As an example, for a simulation in which the number of segments is increased from 1 to 25, the errors in the signal delay computed with the three investigated interconnect models are referenced to the case of an $RLMC$ model with the largest number of segments, considered to be the most precise model in this case. Now, if a maximum value of 1 % for these errors is specified, it can be shown that 7 segments provide sufficient accuracy and may be used for simulation. Because there is not much additional computational overhead involved and the above errors can be further reduced, 10 segments per wire of 1000 µm length have been chosen for the scenarios discussed. In this case, the maximum observed error is about 0.3 % [255].

Nonetheless, the previously discussed figures of merit that have been proposed for line inductance cannot be directly applied when taking into account inductive coupling. Therefore, SPICE [197] simulations have been performed that allow to compare three cases of modeling an interconnect line, namely: a distributed RC line, a distributed simple RLC line without mutual inductances, and a distributed full $RLMC$ line including inductive coupling.

2.3.2 Crosstalk

The electromagnetic fields surrounding each interconnect wire interact with each other and induce undesired signals in all the neighboring lines. When dealing with this unintended interference designers refer to *crosstalk* or *signal integrity*. Crosstalk can have mainly two origins: capacitive and inductive. The capacitive coupling effect is a short-range effect as only the mutual capacitances between adjacent bus lines have a significant influence on crosstalk. On the contrary, mutual inductance decays only slowly with bus-line spacing making the inductive effect a long-range phenomenon [59, 213, 228]. The inductive far-coupling effect can be observed in **Fig**. 2.10, which depicts inductive and

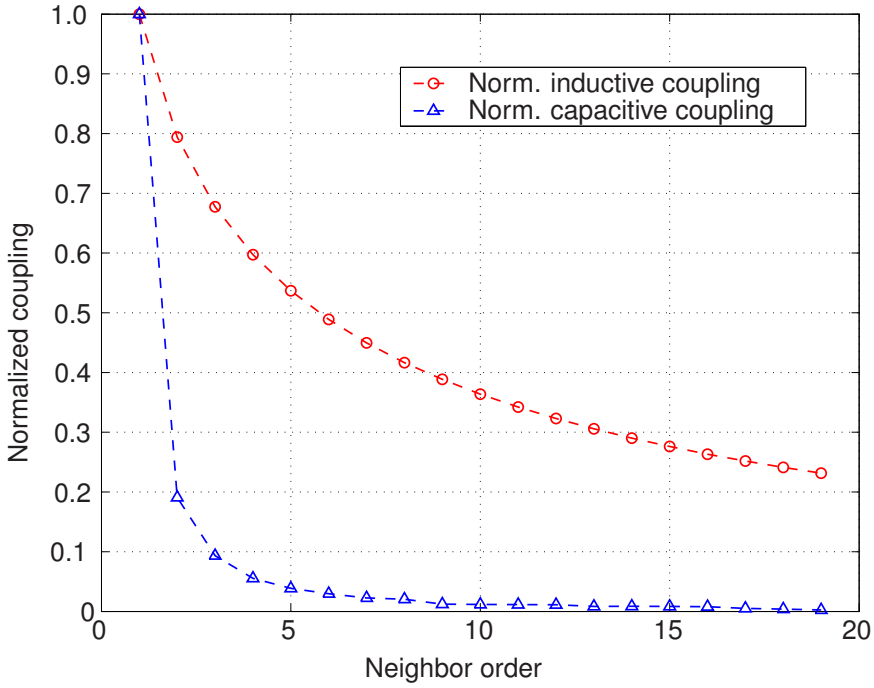


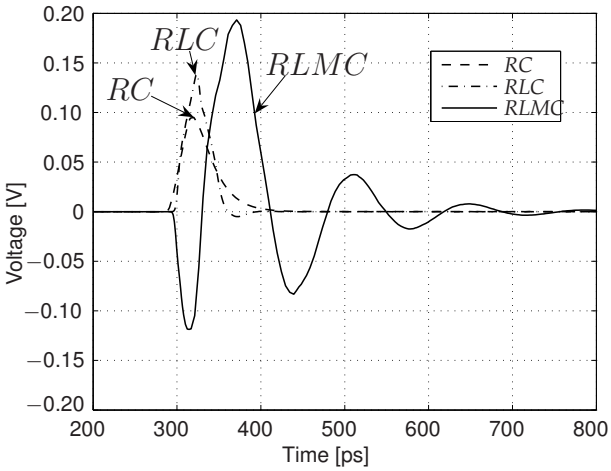
Fig. 2.10: Normalized inductive and capacitive coupling versus neighbor order

capacitive coupling between the first and all other lines of a 20-bit wide signal bus for the distributed $RLMC$ model The figure clearly shows the rapid drop in capacitive coupling between neighbors of higher order compared to inductive coupling. Therefore, spacing is not an effective technique to reduce inductive coupling.

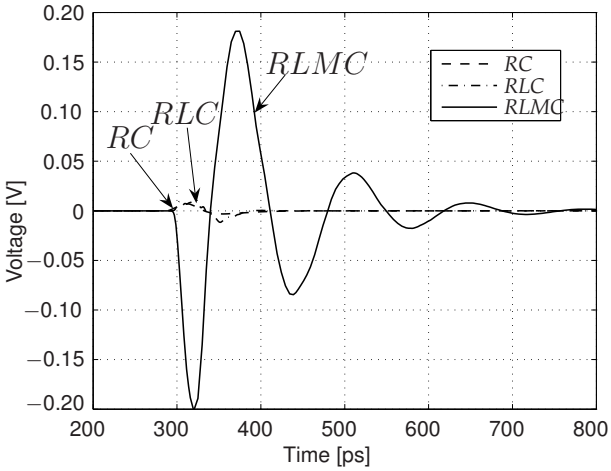
Fig. 2.11 shows simulations of a 5-bit signal bus with four lines held at ground and the remaining neighboring line switching from 0 to 1. The voltage at the far end of the quiet first line is then plotted as a function of time and order of neighboring wire for the three interconnect models investigated. The main conclusion that can be drawn from **Fig**. 2.11 is that the simple RLC model cannot accurately estimate crosstalk. Additionally, it can be seen that the influence of capacitive coupling is comparable to that of inductive coupling when the first-order aggressor toggles. However, in the case of a toggle in the fourth-order neighbor, the voltage glitch predicted by inductive coupling for several transition activity patterns may be even more than one order of magnitude larger than the one predicted by the RLC and RC models. Further details about crosstalk noise for quiet and switching lines can be found in [29,30].

2.3.3 Signal Delay

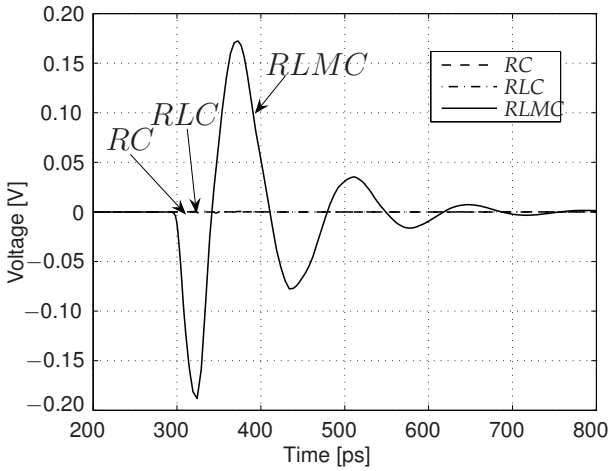
Fig. 2.12 shows two particular cases from the set of simulations results. The waveform of the voltage at the far end of the third signal wire is plotted against time for all three interconnect models.



(a) Transition from ″00000″ to ″01000″

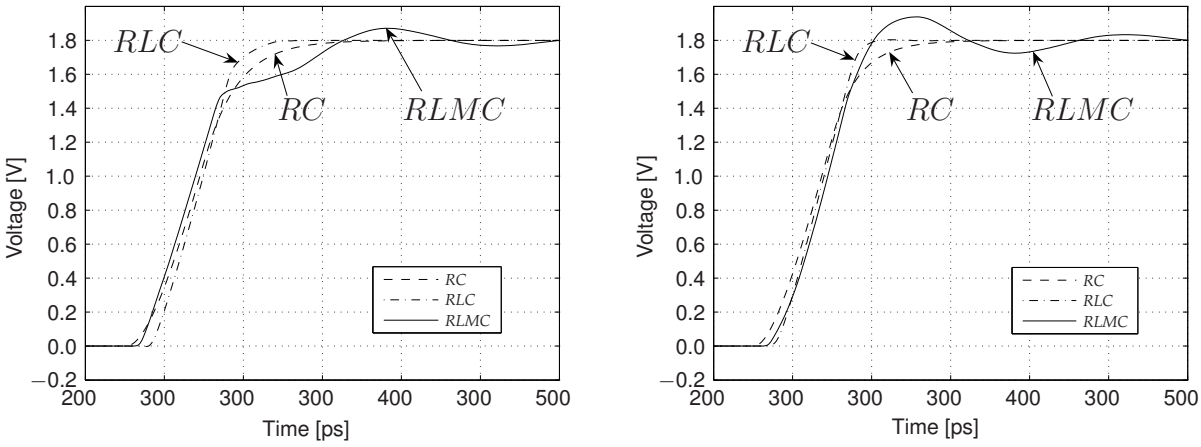


(b) Transition from ″00000″ to ″00100″



(c) Transition from ″00000″ to ″00001″

Fig. 2.11: Crosstalk at the far end of the quiet first line for a toggling in neighbors of different order with an input signal rise time of 25 ps



(a) Transition $0{\downarrow}{\uparrow}{\downarrow}0$ (b) Transition $00{\uparrow}00$

Fig. 2.12: Effects of switching patterns $0{\downarrow}{\uparrow}{\downarrow}0$ and $00{\uparrow}00$ on signal delay and rise and fall times in the third line for RC, RLC, and $RLMC$ modeling

The inclusion of line inductance brings to light not only overshoots and undershoots, which introduce large crosstalk noise on neighboring lines, but also the increase of signal delay [30,109,203]. Disregarding line inductance generally results in underestimating the delay [109]. Nevertheless, as seen in **Fig**. 2.12, the propagation delay may increase or decrease compared to the RC model prediction. Moreover, those delays are strongly correlated to the input data pattern [25,192,213] and models are required at higher levels of abstraction to include these effects. For this purpose, a linear pattern-dependent delay model which accurately predicts signal delay in both inductively and capacitively coupled VDSM interconnects is proposed in **Chap**. 4.

Cao et al. concluded in [25] that when taking into account inductive coupling, worst case delay and noise are dominated more by the switching pattern $\uparrow\downarrow\uparrow\downarrow\uparrow$[2] than of the $\downarrow\downarrow\uparrow\downarrow\downarrow$ one. Furthermore, Tu et al. showed in [192] that the former switching pattern becomes the worst-case scenario with increasing wire capacitance. However, for smaller coupling capacitance the worst-case pattern was reported to change to $\uparrow\uparrow\uparrow\uparrow\uparrow$. Simulation results showed that, for the chosen simulation settings, the best-case switching patterns for the RC model are the worst-case patterns for the $RLMC$ model and vice versa. Depending on the switching pattern, the simple RLC model is closer to either the RC or $RLMC$ model. The signal delay is predicted by the RC model with an error varying between –55.1 % and 78.8 % for $l = 500$ µm and between –62.2 % and 101.6 % for $l = 1000$ µm with respect to the $RLMC$ model.

In the case of capacitive coupling, an aggressor transition in the opposite direction increases the total capacitance the victim has to charge, and the transition is thus slowed down. On the contrary, in inductively coupled lines, an aggressor transition in the same direction induces a current flowing in the opposite direction to the one in the victim. Consequently, the effective current decreases and the delay increases.

[2] $\uparrow$, $\downarrow$, 0, and 1 denote low-high, high-low, quiet on zero, and quiet on one transitions, respectively

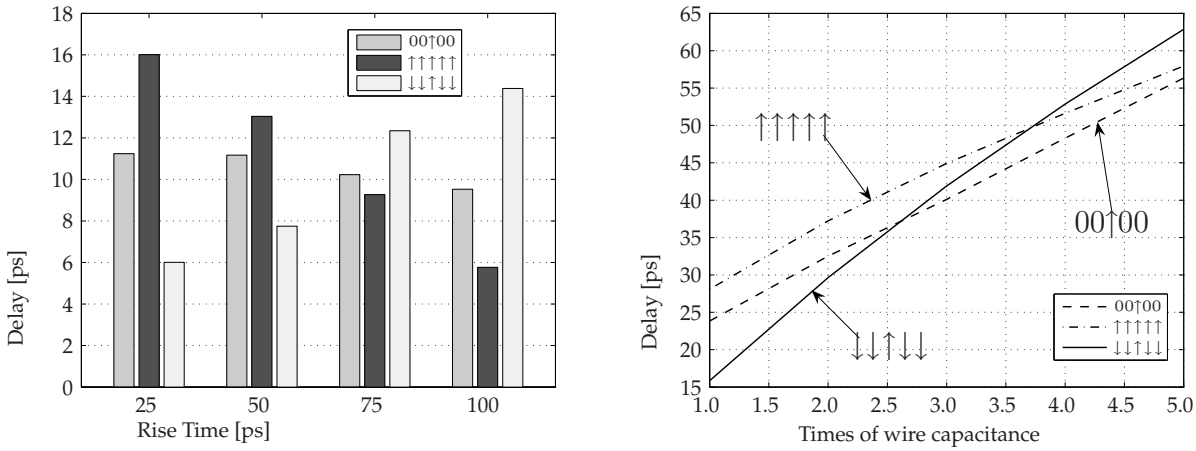
Tab. 2.3: Output rise times for line 3 for input rise times of 50 ps

Transition	RC	RLC	$RLMC$
00↑00	54.6 ps	41.1 ps	46.5 ps
0↓↑↓0	61.4 ps	43.7 ps	89.5 ps
↓↓↑↓↓	60.6 ps	43.2 ps	11.0 ps
↓↑↑↑↓	46.9 ps	38.2 ps	41.6 ps
↑↓↑↓↑	62.3 ps	44.2 ps	50.9 ps
↑↑↑↑↑	47.2 ps	38.3 ps	32.4 ps

Fig. 2.13 a) shows a bar plot of the signal delay for three different values for the input rise time. The three patterns depicted were chosen based on the following reasoning. The first pattern, 00↑00, included as a reference, is considered to be a "neutral" or "nominal" case which does not have a significant impact on signal delay and is always bounded by the delay incurred by the two other patterns. The second pattern, ↑↑↑↑↑, is the best case pattern for the signal delay predicted by the RC model, but the worst case for a highly inductive $RLMC$ interconnect system. The opposite holds for the switching pattern ↓↓↑↓↓, which is the best case for the $RLMC$ model, but the worst case for the RC model. For the sake of completeness, it is also important to mention here, that the value of the far-end load capacitance may significantly influence the inductive behavior of the line as indicated in **Sec**. 2.2.

Because inductance effects generally become less important with decreasing signal frequencies, one would expect a change in the worst case switching patterns for the $RLMC$ model with varying frequencies. This situation is indicated in **Fig**. 2.13 a) where one can observe a decrease in the signal delay caused by the pattern ↑↑↑↑↑ and an increase in the signal delay due to the pattern ↓↓↑↓↓ with increasing rise times and, thus, decreasing significant frequencies. A different approach to show this change in worst-case patterns for the $RLMC$ model is indicated by **Fig**. 2.13 b). In this figure, the input rise time is held at a constant value while the ground and coupling capacitances in the netlist used for SPICE simulations are successively increased from their original values up to five times that value. On the one hand, the expected general increase in signal delay with increasing capacitances can be observed. On the other hand, one can see that the delay caused by the pattern ↓↓↑↓↓ takes over the delay caused by ↑↑↑↑↑ at a certain point, thus clearly showing the greater impact of capacitance than inductance on timing from this point onward. The technological reason behind such an increase in capacitance, particularly in the case of coupling capacitance, is the aforementioned interconnect scaling. Because scaling has a more significant influence on the lateral dimensions of interconnects, the aspect ratio between adjacent signal wires tends to increase and, thus, does the capacitive coupling between these wires.

Previous work showed that when including in the employed interconnect models only line inductances, the rise and fall times of the signal waveforms improve as the inductance



(a) Bar plot illustrating the worst case pattern change with increasing rise time

(b) Delay versus wire capacitance for different worst-case signal patterns

Fig. 2.13: Effects of rise time and wire capacitance on worst case switching patterns and signal delay for the $RLMC$ model

effects increase [109]. Nevertheless, as also shown in [213], when taking into account the coupling inductance, the statement does not hold. The rise time has been computed as the difference between the time instances when the signal reaches 90 % and 10 % of the final value, respectively.

Tab. 2.3 and **Fig**. 2.12 show opposite effects with respect to different lines in the case of the same data toggling context. For instance, in the first case, the rise time in line 3 is predicted with an error of 17.42 % by the RC model, while in the second case, the introduced error is –31.40 %. Thus, rise and fall times (and hence power consumption induced by short-circuits) strongly depend on the data toggling pattern. It is important to add that RLC models generally fail to accurately predict rise and fall times and introduce significant errors.

2.3.4 Power Consumption

The switching component of power dissipation corresponds to the amount of energy required for completely charging the parasitic capacitors. However, this is true only when the transient state is over. Otherwise, the voltage on the capacitors is not necessarily settled, there is still a current flowing through the inductances when the next transition occurs, and therefore some energy is stored in these elements. Simulations showed, that in the case of an $RLMC$ interconnect model, some differences in the switching power dissipated exclusively in the lines appear. Nevertheless, those discrepancies are getting important only when switching occurs very fast. In those situations, the performed simulations showed that the signals are degraded to such an extent which makes them unacceptable anyway. In brief, it can be said that switching power consumption does not depend on inductive effects, but only on the ground and coupling capacitance [228].

For the purpose of analyzing the power consumption in interconnects, inductive effects can be neglected and only the coupling capacitances must be taken into consideration. Basically, when a toggling occurs, more capacitance needs to be charged and discharged than just the ground capacitance because of the capacitive coupling. However, depending on the toggling or non-toggling on the line itself and on the first order neighbors, a line driver could consume a maximum of energy for charging four times the coupling capacitance or receive an energy equivalent to charging two times the coupling capacitance. As mentioned in [228], the power consumption due to capacitive coupling in non-synchronously toggling interconnects is slightly smaller because of the induced dynamic delay.

The switching component of power consumption is independent of the rise and fall times of the input waveforms. However, direct current paths between V_{dd} and ground appear exactly during the rise and the fall of input signals. As indicated in **Sec**. 2.1, the short-circuit power consumption is directly proportional to the rise and fall times. For interconnects modeled only by line inductances, previous work showed that the rise and fall times of signal waveforms improve as the inductance effects increase [109]. Nevertheless, when taking into account the coupling inductance, the statement does not remain true. As previously mentioned, **Tab**. 2.3 and **Fig**. 2.12 show opposite effects with respect to different bus models in the case of the same data toggling context. Because of important overshoots and undershoots, simulations proved that the overall short-circuit power consumption does not necessarily have to decrease, as spurious short-circuit currents can appear whenever a sufficiently big spike opens the complementary transistors [213]. Thus, it can be concluded that the short-circuit power consumption strongly dependends on the capacitive and inductive coupling as well as on the input data patterns.

In order to assure high performance, long resistive interconnects are driven by repeaters. Those repeaters are generally large gates and are responsible for a notable part of the total power consumption. When not taking into account the mutual inductances, as line inductance effects increase, the optimum number of repeaters for minimum propagation delay decreases [109,70]. Moreover, fewer and smaller repeaters result in a significant reduction of the dynamic power consumption induced by those buffers. Nonetheless, as previously seen, the worst case signal delay predicted by an $RLMC$ interconnect model degrades dramatically compared to the signal delay resulted in simulations of an RLC model. Thus, in inductively coupled on-chip interconnects, the optimum number of repeaters is generally higher than the number predicted by an RLC model and the expected savings in area and power consumption are therefore too optimistic.

2.3.5 Influence of Process Variations on Interconnect Parameters

In the following, the effects of process variations on interconnect parameters is analyzed by modeling width, pitch, and thickness as Gaussian distributions. Using the same wire topology as in the previous section, uncorrelated sets of 1000 Gaussian distributed values

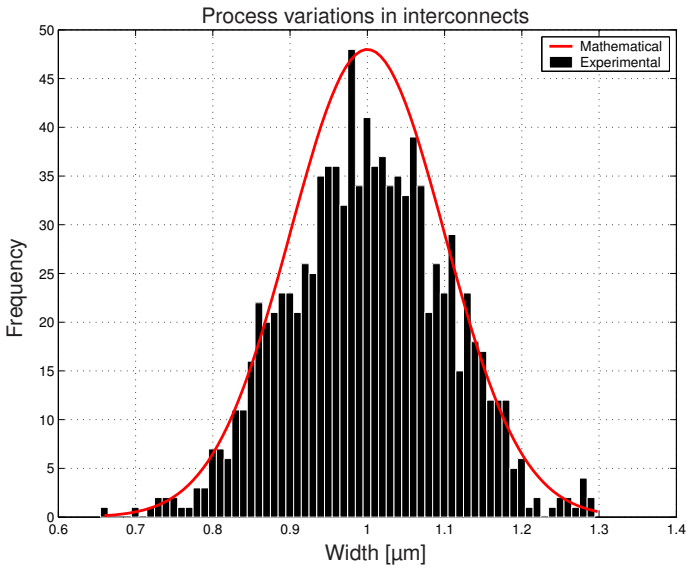


Fig. 2.14: Process variations modeling for width (w_{mean}=1 µm) modeled with 1000 normally distributed random values. Thickness and pitch were modeled in the same way (t_{mean}=1 µm, p_{mean}=3 µm)

for pitch, width, and thickness have been generated. **Fig**. 2.14 illustrates such a distribution for the wire width. The thousand different sets of values were used in field solver simulations with FastCap and FastHenry to obtain the interconnect parameters depending on process variations. Example plots are shown in **Fig**. 2.15.

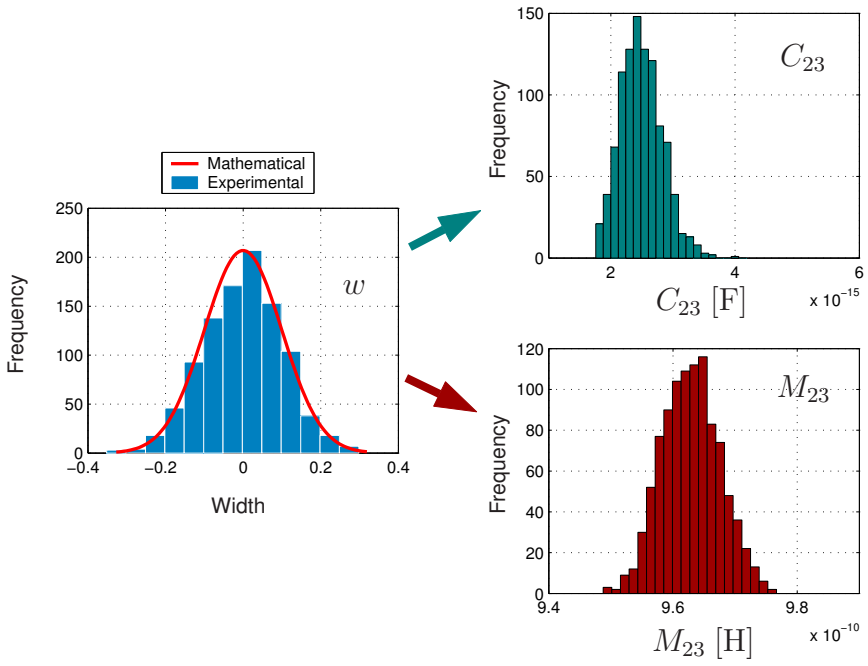


Fig. 2.15: Influence of process variations on interconnect parameters

Fig. 2.15 shows the distribution of a geometric parameter on the left side and the effect on coupling capacitance C_{23} and coupling inductance M_{23} on the right side. Because coupling capacitance is proportional to the inverse of the distance between wires, its distribution is not Gaussian. Likewise, (mutual) inductance is a nonlinear function of interconnect dimensions and therefore a non-Gaussian distribution. It is important to notice that process variations affect the inter-wire capacitance much more than the mutual and self inductance. Thus, especially the variations in capacitance must be accurately taken into consideration. In **Chap**. 4, the proposed pattern-dependent delay model is extended to encompass also process variations.

2.4 Summary

Based on a rigorous analysis of the impact of technology scaling on interconnects structures, this section discussed first the effects of very deep sub-micron technologies on performance and power consumption. It has been shown that since global, intermediate, and local wires are scaled down following distinct rules, different effects will prevail in each case. Afterwards, based on the PEEC method several interconnect models of increasing complexity have been presented. Moreover, the characterization of driving gates and the interfacing thereof with interconnect models has also been addressed.

Distributed models including also capacitive and inductive coupling have been chosen for analyzing their impact on crosstalk, delay, and power consumption. The most important outcome is that all those parameters are pattern-dependent, and in order to be able to perform an accurate high-level analysis, efficient delay models are required that can cope with that pattern dependency. Moreover, it has been shown that capacitive and inductive coupling have antagonistic effects on delay and that under normal operating conditions, inductive coupling does not influence the switching component of the dynamic power consumption. Nonetheless, inductive effects modify slews, affecting therefore the short-circuit component of the dynamic power consumption which represents usually a very small fraction of the total dynamic power consumption in optimized interconnects. Furthermore, it has been shown that process variations have a significant influence on interconnect model parameters and that those variations have to be taken into account in any accurate delay model.

Chapter 3

State-of-the-Art in Interconnect Optimization

Contents

In the previous chapter, it has been pointed out that interconnect parasitics have a growing impact and pose an increasingly severe pressure on the design process. Resistive, capacitive, and inductive effects together with increased influenced process variations may cause important signal perturbations and crosstalk affecting thus overall performance, reliability, power consumption and thermal requirements.

The abovementioned issues can be partially or even sometimes completely mitigated by means of different techniques available at different abstraction levels. This chapter discusses – albeit not exhaustively – the benefits and drawbacks of several such optimization

methods applicable at different levels: technology, layout and routing, circuits, architectural and system. First, traditional and more radical advances and paradigm shifts in technology are enumerated. Besides better interconnect materials, one can envisage integrating on-chip optoelectronical, molecular, or carbon nanotubes for designing efficient on-chip communication structures. Secondly, several measures like shielding by means of inserting lines or complete ground planes, wire sizing and shaping can be considered at the layout and routing level. Third, a multitude of optimization opportunities like buffer or booster insertion, partitioning and cascading drivers, driver sizing, line termination circuits, precharging circuits, or voltage scaling are available at circuit level.

As this thesis deals with optimizing delay and power at high levels of abstraction by means of signal encoding schemes, the focus is put on architectural and system level optimization methods. Here, optimization solutions span a vast domain from novel interconnect-oriented architectures up to more abstract communication schemes (protocols, fault-tolerance, signal encoding) and early high-level interconnect planning design methods.

3.1 Technological Level

The worst coupling case is fundamentally determined by wire and dielectric materials as well as their (minimum) sizes and manufacturing process. As mentioned by the SIA Roadmap in [67], there are two main trends in interconnect optimization. For the near term which refers to technologies larger than 32 nm, the most difficult challenges are related introducing new materials that meet the wire conductivity requirements and reduce the dielectric permittivity. Nevertheless, for the long term, i.e. technologies lower than 32 nm, performance requirements will not be satisfied anymore with traditional scaling and the solution will derive only from innovative methods like optical, radio frequency, molecular, carbon nanotubes, or vertical integration combined with accelerated efforts in packaging and design [67].

One of the most challenging interconnect-related predictions ever made by the ITRS [66, 67] has been the lowering of the dielectric constant and the industry transition has taken longer than any other previous roadmap predictions. In order to achieve even lower K values, porosity can be added. However, integrating porosity is expected to be even more problematic as all possible solutions provide a lower reliability due to their lower chemical integrity and thermal expansion coefficients that stress metalization.

Wire resistance of polysilicon and wires, and thus interconnect delay, has been reduced by the introduction of copper and silicides. Nevertheless, copper, like some low-K materials, does not solve the fundamental problem of long wire delay and represents just a temporary solution for a couple of technological generations [141]. Thus, meeting the conductivity requirements remains an essential challenge also in the near term. Moreover, difficult challenges in the near term regarding interconnects are mainly related to

reliability, manufacturability, integrating interconnect structures with new materials and processes, process variability, and three dimensional control of interconnect features [67].

Copper and low-K are expected to continue to find applications in future technology generations, however, novel and more radical interconnect solutions will be eventually required to cope with the increasing challenges. The main long term challenge identified by the ITRS is related to identifying solutions that address global scaling issues [67]. It is widely accepted that by applying the traditional scaling scheme, performance requirements will no longer be satisfied. Therefore, unconventional and radical interconnect solutions beyond copper and low-K have to be defined.

Photonic or optical interconnects have been considered as potential candidates for solving global interconnect issues. However, they have gained an increasing interest and started to be regarded as a viable solution in the future only after significant progress has been achieved in the last years. In order to be competitive with classical interconnects, optical electronics must provide first high-speed, low-power communication devices at small sizes and manageable costs. Moreover, those devices have to be compatible with CMOS technologies [37].

In three-dimensional integration, active layers are stacked one upon another. The main benefits of three-dimensional integration are the high achieved density and the reduction of interconnect length, i.e. improvement of interconnect performance. The fundamental issues are reliability and manufacturability of 3D stacks and strict thermal management, i.e. heat dissipation flow [37, 67].

Other aggressive and radical ideas include on-chip RF/microwave, cooled superconductors, carbon nanotubes, and molecular interconnects [68]. However, those methods are in their early stages of development and issues like technology maturity and mainstream transfer have still a long way to go.

3.2 Layout and Routing Level

Crosstalk depends not only on signal characteristics like rise time but also on the geometry of the lines. Layout techniques integrated in place and route tools can be used to reduce crosstalk and mitigate the undesired induced effects.

3.2.1 Increased Metal Separation and Shielding

When the coupling capacitances between bus lines increases and gets comparable to or larger than the ground capacitance, crosstalk noise at the input of the subsequent gates can produce undesired glitches which at their turn can generate malfunctions. Spacing or increased metal separation is an anti-crosstalk technique in which a trade-off between the total bus area and the maximum allowed crosstalk is realized. Wires are spaced as distant

as possible from each other and the coupling capacitance are reduced as much as possible. Additionally, inter-wire coupling can be reduced by making sure that no two lines are laid out parallel or next to each other for longer than a maximum length. However, due to the fact that the mutual inductance decays much slower than the coupling capacitance, spacing is not an efficient method if inductive effects are significant.

Multilayer interconnections emerged as a viable method for efficiently connecting transistors on-chip [6]. As a great amount of chip area is occupied by interconnects and because the average interconnection length is inversely proportional to the total number of layers, multilayer metaling reduces die area and improves interconnect performance. However, when inductive effects cannot be neglected, current return paths are very difficult to predict in higher metal layers if no dedicated ground planes or lines exists. Screening techniques consists of adding an extra ground or V_{dd} line close to the lines affected by crosstalk. This way, crosstalk is reduced as a close return path is provided. Shielding refers to inserting a ground or V_{dd} line between two signal lines. Even though shielding is an area-expensive technique, it makes the behavior of interconnects in terms of crosstalk and delay a lot more predictable [141]. The fundamental idea is that the smaller the loop inductance, the smaller the $L\frac{di}{dt}$ noise.

Shielding represents at first sight a waste of interconnection resources but in some applications it may be preferable to having many reference planes. Furthermore, ground planes slightly increase C_g, but C_c remains in general unmodified while inductive effects can be significantly decreased. Moreover, simultaneous shield insertion and net ordering (SINO) for RLC nets has also been addressed in [58]. Several algorithms have been proposed based on Greedy techniques, Graph-Coloring and Simulated Annealing. However, due to the long range of current return paths, shielding is capable of screening only part of these signals current return paths, and thus depending on the rise times and wire geometry it might eliminate the inductive coupling only partly [111, 112].

3.2.2 Wire Sizing, Wire Splitting, and Interconnect Routing

Although wire sizing is a very efficient technique to optimize delay in non-coupled RC lines, it has reduced applicability for delay improvement in coupled RC and inductive lines. Cheng et al. showed in [30] that low-frequency self-inductance does not decreases by more than 10 % with the doubling of the width, and that the reduction at high frequencies is even smaller due to skin and proximity effects. At high frequencies, because of the skin and proximity effects, the majority of the currents are conducted around the corners of the wire [30]. Wires wider than about two skin-depths do not really result in a larger high-frequency resistance reduction. Therefore, lines have to be split in order to reduce line reactances as for instance, splitting a wire characterized by significant skin and proximity effects in N parallel wires about two skin-depths width each could reduce the reactance by N times. Inductive effects can be further reduced by combining wire splitting with insertion of dedicated ground lines. This is especially interesting for the

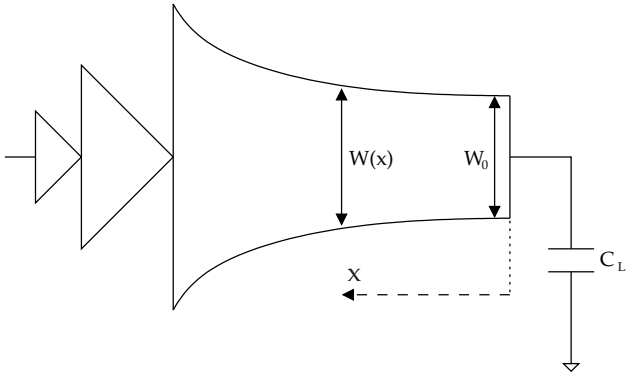


Fig. 3.1: Wire shaping or tapering (after [41])

scope of this thesis, as signal encoding can be regarded at lower levels of abstraction as an enhanced combination of wire splitting, spacing, and/or dedicated ground/power line insertion.

It has been shown in [41] that the width of an inductive interconnect affects the power consumption. As the ratio between self-inductance and resistance increases with the wire width, the signal transition time is also reduced. Thus, the short-circuit power consumption component decreases. However, due to the matching characteristics between line and interconnects, the short-circuit power consumption augments. With increasing wire width, the spacing between lines is reduced which leads to higher coupling capacitances. Therefore, the switching power consumption increases with wire width and an optimum can be found for the total dynamic power consumption.

Wire shaping can improve circuit speed and for RC lines the optimum wire shaping function that minimizes delay is an exponential function [41]. Wire tapering is a technique that increases the wire width at the near-end of the line as shown in **Fig**. 3.1. Thus, the total line resistance is reduced as well as the resistive shielding effect while the line inductance is increased. The propagation delay in RLC interconnects is determined by the relationship between the RC time constant and the LC time constant (time of flight). If the inductive behavior of a line dominates the resistive behavior than the signal propagation delay is determined by the inductive time constant. It has been shown in [41], that the wire tapering function gives the following width, $W(x)$:

$$W(x) = W_0 \cdot \mathrm{e}^{\frac{2L_0C_0}{c}x}, \tag{3.1}$$

where

$$c = \frac{2C_f L_0 l}{W_0} \tag{3.2}$$

and L_0, C_f, C_0 are the line inductance per square, the fringing capacitance per unit length, and the line capacitance per unit area, respectively. The far-end width W_0 can be determined numerically.

As mentioned in [141], any approach helping to reduce the wire length is bound to have a significant impact. It has been shown that routing also in the diagonal direction –

the so-called 45° routing – achieves an important reduction in wire length in comparison with the classical Manhattan-style routing. Nevertheless, inductive coupling becomes more of an issue as it cannot be generally neglected between neighboring metal layers containing non-orthogonal lines. Furthermore, crosstalk effects can be controlled by using predefined wire structures, so-called regular fabrics. Thus, designers can ensure through conservative design that the requirements are satisfied, however, with a cost in flexibility.

3.3 Circuit Level

In the following, several circuit level techniques that reduce crosstalk, improve signal delay, and decrease power consumption in interconnects are reviewed. The most widely used are buffer (or booster) insertion, anti-crosstalk line terminations, precharging circuits, differential signaling, etc.

3.3.1 Line Terminations

Line terminations or line matching is a technique that has been extensively used in printed circuit board (PCB) design in order to reduce transmission line effects with minimal performance sacrifice [30]. The terminations are inserted at the far end in order not to affect

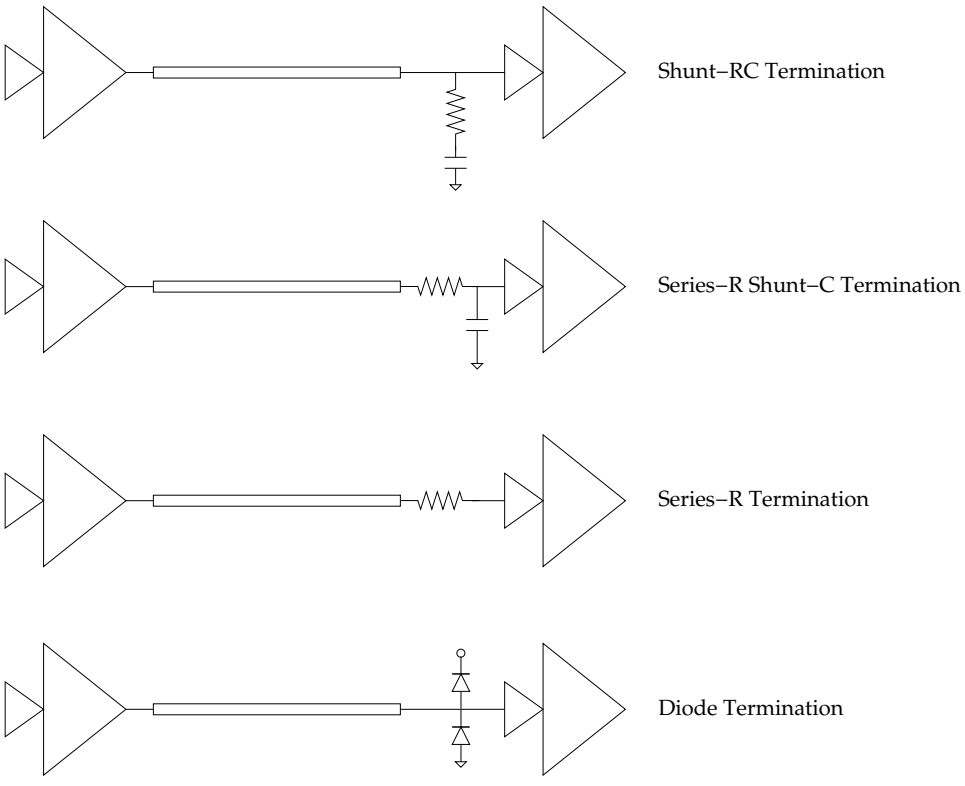


Fig. 3.2: Line terminations (after [30])

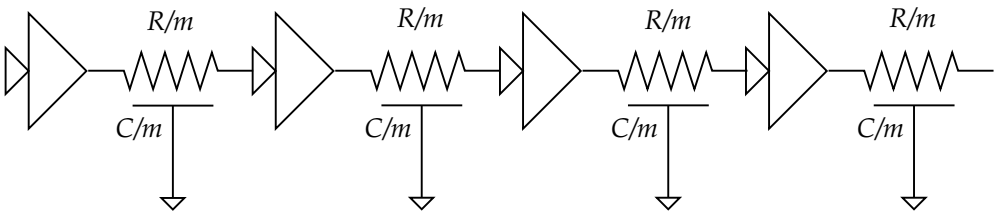


Fig. 3.3: Reducing RC delay via buffer insertion (after [141])

the performance by slowing down switchings. Basically, there are four possible terminations as illustrated in **Fig**. 3.2: Shunt-RC, Series-R Shunt-C, Series-R, and Diode.

In [30], the performance of the abovementioned terminations has been analyzed and compared. The diode termination has been rapidly discarded due to the high induced power consumption. Thus, it has been concluded that compared with the Shunt-RC termination, the Series-R Shunt-C termination provides a more efficient noise reduction at the expense of a higher performance loss. Further, the Series-R termination exhibits the best noise decreasing at the lowest performance loss, and is thus to be preferred in high-speed VDSM technologies.

3.3.2 Buffer Insertion

Traditionally, buffers (or repeaters) have been inserted in long interconnects in order to reduce the RC delay, which is a square function of the total length (see **Fig**. 3.3). However, buffer insertion is a technique that efficiently reduces inductive effects by shortening the initial current paths. This results in smaller current loops, and thus decreased inductive crosstalk. Buffer insertion typically reduces crosstalk noise but sometimes degrades the performance due to the additional delays of the inserted buffers and there is clearly an optimum delay that can be achieved by means of repeater insertion.

Considering that the buffers exhibit the same delay t_b, the optimum number of repeaters n_b for an RC line of PUL resistance R, PUL capacitance C, and length l can be derived as given in [141]:

$$n_b = l\sqrt{\frac{0.38RC}{t_b}} = \sqrt{\frac{t_w}{t_b}}, \tag{3.3}$$

with a corresponding minimum wire delay:

$$t_p = 2\sqrt{t_w t_b}, \tag{3.4}$$

where t_w represents the delay of the unbuffered line. The optimum is obtained when the delay of the buffers is equal to that of the wire segments.

The buffer propagation time depends nevertheless on the load capacitance and therefore, it is of utmost importance to size the repeaters in order to efficiently decrease the total delay. By denoting with R_d and C_d the resistance and input capacitance of a minimum-sized buffer, respectively, and employing the Elmore delay model, the following values

for buffer number, buffer size (s_b), and minimum wire delay are obtained:

$$n_b = l\sqrt{\frac{0.38RC}{0.69R_dC_d(1+\gamma)}} = \sqrt{\frac{t_w}{t_b^{(1)}}}, \tag{3.5}$$

$$s_b = \sqrt{\frac{R_dC}{RC_d}}, \tag{3.6}$$

$$t_p = \left(1.38 + 1.02\sqrt{1+\gamma}\right) l\sqrt{R_dC_dRC}, \tag{3.7}$$

where γ denotes the ratio between the intrinsic output and input capacitances of the buffer, and $t_b^{(1)} = 0.69(1+\gamma)R_dC_d$ represents the delay of an inverter for a fan-out equal to one [141]. Further, there exists an optimal or critical length, $l_c = l/n_b$, for a given technology and a routing layer, and the delay, t_c, of a segment of critical length is independent of the routing layer:

$$t_c = \frac{t_p}{n_b} = 2\left(1 + \sqrt{\frac{0.69}{0.38(1+\gamma)}}\right) t_p = 2\left(1 + \frac{1.3475}{\sqrt{1+\gamma}}\right) t_p. \tag{3.8}$$

Therefore, inserting buffers is useful only if the critical length is at most half of the wire length.

Ismail and Friedman showed in [70], that the formulas derived for RC lines can be generalized for lines exhibiting self inductance as follows:

$$n_b = l\sqrt{\frac{RC}{2R_dC_d}} \cdot \frac{1}{\left[1 + 0.18\left(\frac{l}{2\xi\sqrt{\frac{RC}{R_dC_d}}}\right)^3\right]^{0.3}}, \tag{3.9}$$

$$s_b = \sqrt{\frac{R_dC}{RC_d}} \cdot \frac{1}{\left[1 + 0.16\left(\frac{l}{2\xi\sqrt{\frac{RC}{R_dC_d}}}\right)^3\right]^{0.24}}, \tag{3.10}$$

where ξ represents the damping factor of an RLC line as defined in **Sec**. 2.2. It is to be noticed that the optimal number of buffers in an RC line has been approximated as:

$$n_b = l\sqrt{\frac{RC}{2R_dC_d}} \tag{3.11}$$

When the damping factor tends to infinity, the right factors of the above formulas tend to one, and the case of a purely RC line is obtained. Further, it has been argued that the traditional quadratic delay of RC lines tends to a linear dependence with increasing inductive effects. Nevertheless, it must be added that for short to medium segments, inductance increases delay. Moreover, as intrinsic gate delay steadily decreases with technological advances, long lines can be split into many short segments in order to achieve the optimum delay. This means, that with each extra introduced buffer, the segment length (and thus inductance) is decreased.

3.3.3 Advanced Signaling Techniques and Driving Circuits

The most common way to decrease switching-induced delay when driving large loads is to increase the driving transistor sizes and thus the average switching current. In the last years, several other advanced methods and circuits for driving long on-chip interconnects emerged, for instance differential signaling, boosters, or reduced-swing circuits [204].

Because repeaters are responsible for increasing area, power, and design resources, and due to the fact that they are inherently limited in how much they can improve the performance, a novel circuit technique called *booster* has been presented in [124] for driving long on-chip interconnect. Boosters have the advantage of being bidirectional and providing a low impedance termination to improve signal integrity. Moreover, they are typically inserted three times less frequently than repeaters for optimal performance which also results in a reduced total power consumption. Unlike traditional repeaters, boosters possess a floorplan-friendly characteristic as they are relatively insensitive to placement variation. Placing a nonoptimal number of boosters does not hurt the overall performance which makes them very attractive solution for driving long noisy wires.

Low-swing or reduced-swing circuits reduce the signal swing at the driver output. A generalized scheme for reduced-swing interconnect circuits is illustrated in **Fig**. 3.4. The delay is reduced linearly with the voltage swing, while current is also reduced [141]. Besides an improved performance, low-swing circuits significantly decrease the dynamic power consumption which is of paramount importance especially when driving large capacitive loads. The main drawbacks of low-swing signaling are reduced signal integrity and noise margins. Furthermore, for low-swing on-chip data transmission to work properly, circuits similar to sense amplifier have to be employed. Low-swing circuits can be static or dynamic (precharged), single-ended or double-ended (differential) [141].

The main idea behind differential signaling is to transmit both the signal and its complement obliging thus the neighboring lines to toggle in opposite directions. Consequently, the capacitive crosstalk in the sensitive low-swing networks is reduced, and even more importantly, the inductive noise is also considerably decreased by providing nearby return paths for fast signals. Again, the technique can be regarded as a low-level signal encoding scheme especially because differential signals are implemented by using

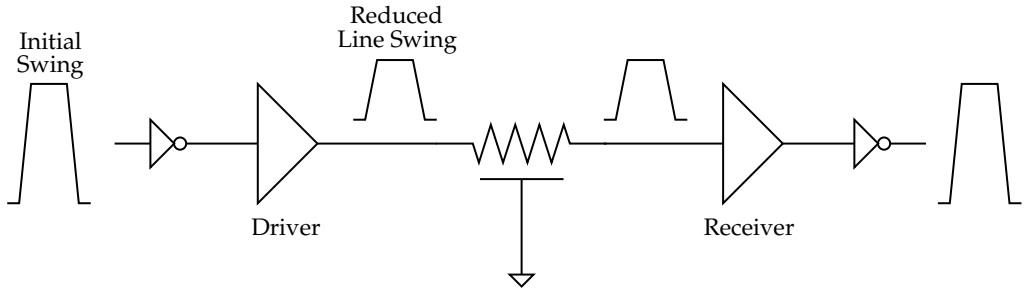


Fig. 3.4: General scheme of a low-swing interconnect structure (after [141])

interleaved redundant signals. Signal encoding is even more promising, due to the fact that in order to reduce capacitive coupling, signal lines and their negated counterpart should not be placed right next to each other [30].

Another method to improve the response of large interconnect structures and not only, precharching can be employed [141]. The benefit of speeding-up the loading of large loads comes however with typical drawbacks of dynamic circuits: leakage, charge sharing and loss. Therefore, improvements like the so-called pulse-controlled driver with sense amplifier have been proposed for circuits in which the capacitive loads are well known [141].

One interesting technique to reduce power consumption in digital CMOS circuits is offered by the so-called adiabatic circuits. Power can be decreased by introducing an inductor in the charging path and thus slowing down the charge transport. Additionally, the inductor can act as an energy storage element, conserving the energy that is normally dissipated during discharging. An LC resonant circuit is formed together with the parasitic capacitances [166, 198]. An essential disadvantage of adiabatic circuits is their poor power-delay product.

While charge recovery is very attractive because of its asymptotically zero energy, it is advantageous only at very low operating rates. It has been argued that voltage scaling is much more efficient in both energy and performance when including the energy consumed in the rail drivers [64].

3.4 Architectural and System Level

With increasing system complexity, chip size, and operating frequencies, solving interconnect problems will be crucial for design engineers. As on-chip communication becomes to be limited by physical constraints like speed of light or thermal noise, interconnect planning, design, and optimization has to be tackled also at architecture and system level where significant optimization opportunities can be exploited. Therefore, interconnect structures emerge as one of the most (if not the most) critical design issue also at high levels of abstraction.

3.4.1 High-Level Interconnect Planning and Optimization

Cong presented in [32] a set of concepts, algorithms, and methods for an interconnect-centric design flow. The design flow allows interconnect design and optimization at every layer of the design process as it consists of three major phases: interconnect planning, interconnect synthesis, and interconnect layout. Moreover, the flow disposes of several interconnect prediction models. The main idea behind the concept is the goal that designer teams should focus only on designs mainly at system level.

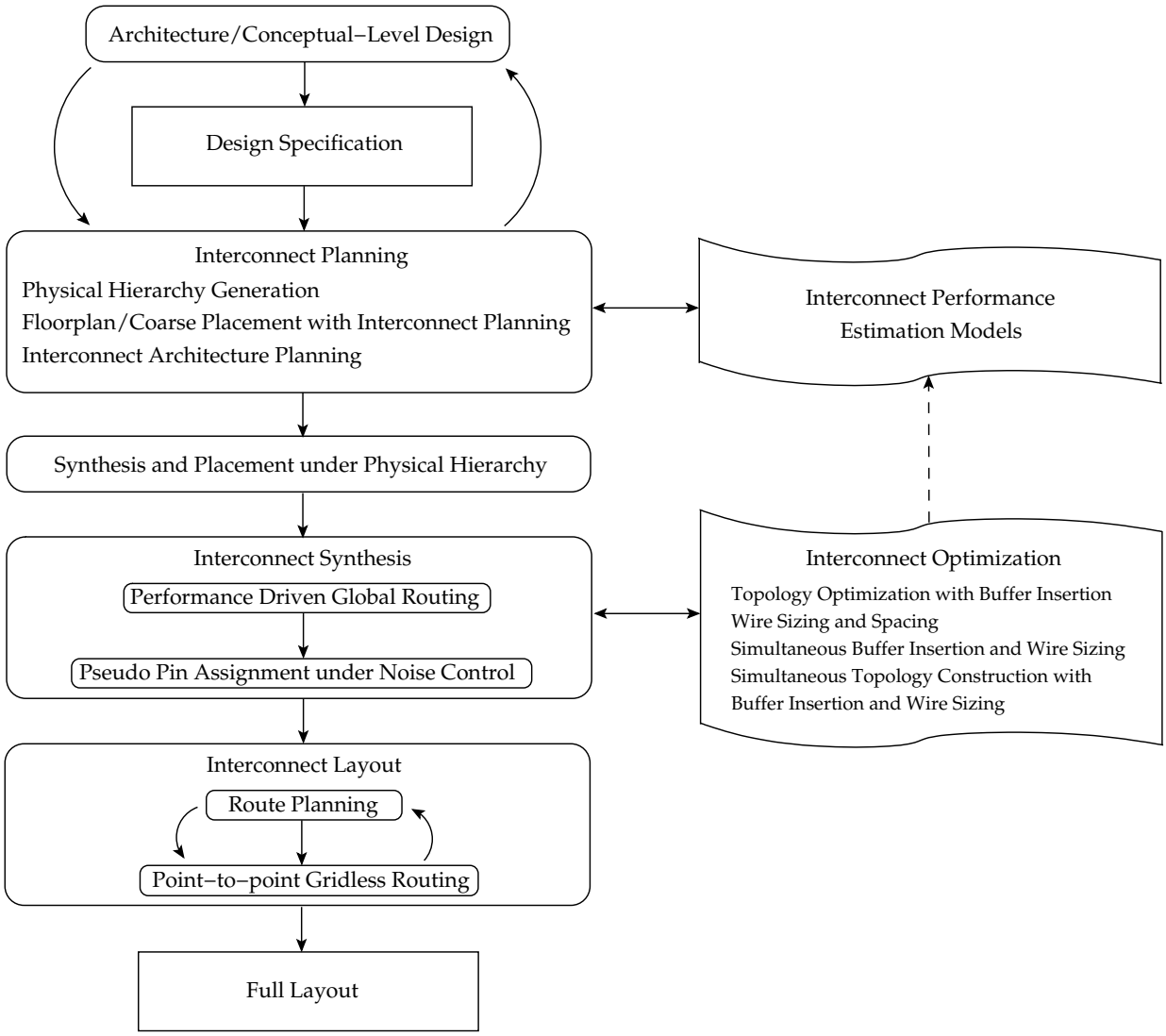


Fig. 3.5: Overview of the interconnect-centric design flow proposed in [32]

Given a specification, the functional hierarchy is transformed into a physical hierarchy during the first phase, i.e. interconnect planning. Additionally, coarse placement and global interconnect planning can be performed simultaneously at early design stages. At this step, iterations with the conceptual level should be done in order to be able to exploit as efficiently as possible optimization opportunities at high levels of abstraction. Secondly, synthesis and placement for each module is performed under the constraints of the resulted physical hierarchy. Afterwards, interconnect synthesis follows including performance-driven global routing along with a set of delay and noise minimization techniques. Finally, interconnect layout is completed by means of a coarse grid based route planning engine and a point-to-point one. Although the abovementioned design flow is yet to be fully implemented, the concepts presented in [32] share the important quality to have highlighted the importance of interconnect-centric design flows and to have partly shown what requirements, constraints, and goals the algorithms of the flow must face.

An essential step of any interconnect-centric design flow in terms of achieving high performance at minimum power and area penalty is the buffer insertion process. Traditionally, the inserted buffers barely influenced the total area and power consumption of a system. However, with rapidly increasing interconnect length, the number of total required repeaters augments dramatically. Consequently, buffers are reported to become a problem at both chip- and block-level [56, 165] as they will eventually be responsible for the majority of the die area and total static power consumption (leakage-induced). Such an explosive increase in repeater number will finally have a profound impact on the design flow as area and power consumption minimization in buffered interconnects must be addresses during the early design phases [24, 30]. In this context, a simultaneous placement and buffer planning for reduction of power consumption in interconnects and repeaters has been proposed in [210] and is discussed in detail in **Chap**. 7.

3.4.2 Interconnect-Centric Architectures

Interconnect optimization can be performed also at run-time and not only at design-time as previously discussed. It is of utmost importance that architectures and system operation should be conceived in such a manner that performance and power consumption can be adapted during operation depending on the work load and required performance. For instance, if at any time during system operation, highest peak communication performance is not required, power can be traded for performance. Such techniques are known as dynamic power-performance management or simply dynamic power management [9].

Even with an efficient repeater insertion, the wire delay cannot be reduced as much as desired. Thus, for large designs it can turn out that the delay in global wires is larger than the clock period. As mentioned in [141, 205], the only possibility to cope with performance-limiting interconnects is at architecture level. The concept of pipelining can be borrowed from improving the performance of long critical path in order to realize the so-called multi-cycle communication [33] or wire pipelining [141] represented in **Fig**. 3.6. While such a technique does not reduce the total wire delay, throughput can be significantly improved as the interconnect is transmitting many signals concurrently. It is to be noticed that pipeline-like interconnection structures have a fundamental impact on general architecture and system organization. In order to rapidly assess the effectiveness of competing interconnect architectures, system level simulation engines based on real and/or statistical data are required [215].

When small signal delays are not required, supply voltage can be scaled down reducing thus power consumption significantly, as the dynamic component of energy consumption is proportional to the square of the supply voltage [45]. Therefore, voltage scaling (or multiple-V_{dd}) is a particularly powerful and interesting method which has to be included in any architecture exhibiting a workload that is variable in time. Moreover, it must be addressed at every level of abstraction of the design process, especially during the early phases as communication planning and synthesis [134, 135, 242].

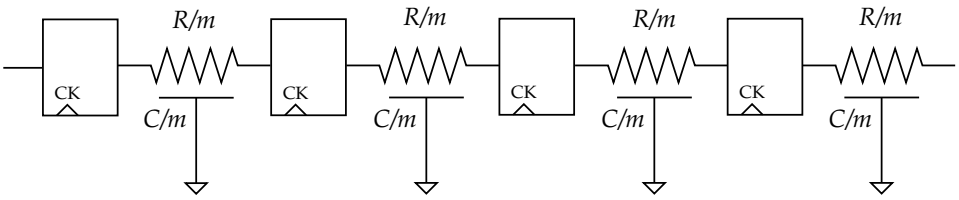


Fig. 3.6: Multi-cycle communication or wire pipelining (after [141])

Another powerful method is threshold voltage scaling (or multiple-V_{TH}) as at higher threshold voltages, driving gates react more slowly at input switchings but exhibit at the same time a significantly decreased leakage current [184]. Hence, multiple-V_{dd} techniques have to be combined with multiple-V_{TH} ones in order to achieve an optimal compromise between performance and total power consumption [182]. Furthermore, in order to reduce the leakage-induced power consumption, idle blocks can be shut down fully or at least partially [9].

Under the previously described scenario in which global and total interconnect lengths increase dramatically, synchronization of future systems will become a difficult if not impossible task to achieve. Hence, a centralized traffic control is getting intractable and determinism of on-chip communications will disappear. Therefore, an on-chip interconnection paradigm involving local synchronous and global asynchronous (GALS) communication might prove the only feasible solution. Abstracting the physical interconnections will offer the possibility to cope with problems like unavoidable data failures on the physical layer or distributed traffic control. Issues like synchronization, scalability, reuse, distributed control, reliability, lead to the idea to employ solutions proposed for efficient interconnecting in large-scale communication and networking systems. Consequently, interconnects should be regarded as communication channels in a more abstract way rather than considering them point-to-point wires. In this context, Benini and De Micheli introduced in [10] the concept of Networks-on-Chip (NoC). Hereby, techniques and methods already developed in networks system theory can be reused, while taking however into account the influence of specific on-chip interconnect characteristics like VDSM effects, power consumption, and crosstalk, to name only a few [72].

3.4.3 Signal Encoding for Power, Crosstalk, and Delay Optimization

A recently proposed and powerful method to improve performance and/or reduce noise induced by crosstalk in interconnects is to avoid worst case patterns by means of signal encoding schemes [141,179]. Thereby, the transmitted data can be modified in such a way that those transitions that spawn large delays or crosstalk are eliminated. Further, data encoding can also be employed in order to reduce power consumption [187,189], an idea which has been proposed more than a decade ago and reached a remarkable maturity, as a multitude of general and also application-specific codes have been proposed in the literature.

As explained in more detail in **Chap**. 4, the worst case switching patterns for delay and power are identical in capacitively coupled buses. There are however two fundamental differences between coding-based performance improvement and coding-based power reduction. First, the latter generally minimizes the amount of worst-case transitions but does not necessarily eliminate them completely. Secondly, with increasing inductive effects, the worst case patterns for delay change and those for power remain unmodified.

The underlying idea behind coding schemes for power is to find a mapping function Ψ_C that transforms the input codeword alphabet into another one that assigns low-power transitions to the most probable switching patterns and low-power states to the most frequent ones. Thus, the dynamic and static power consumption, respectively, are reduced. As the energy consumption is determined by the transitions on the bus, the coding scheme requires information on both received and transmitted codewords as illustrated in **Fig**. 3.7, where v_k and w_k represent the input and output codeword at time k, respectively. The efficiency of the mapping function is strongly related to the amount of information it can handle. However, the complexity of the coding scheme explodes with every new input and there is clearly an optimum between optimizing power consumption and keeping the complexity and the implied associated power consumption of the encoder acceptable. The main goal is to enhance interconnect synthesis tools in order to support an automated – at least partially – implementation of coding schemes [46, 105].

For a bus width B, let the so-called $2^B \times 2^B$ energy cost matrix $\mathcal{E}[w^{(i)}, w^{(j)}]$ be the matrix that completely describes the energy cost associated with the transitions from $w^{(i)}$ to $w^{(j)}$ and the associated states. Considering that the code requires n input and m output codewords for constructing a new codeword, the output codeword at time m is defined as:

$$w_m = \Psi_C[\underbrace{v_m, v_{m+1}, \cdots, v_{m+n-1}}_{n \text{ input codewords}}, \underbrace{w_0, v_1, \cdots, v_{m-1}}_{m \text{ output codewords}}], \tag{3.12}$$

where the initial codeword w_0 is defined as:

$$w_0 = \Psi_C[v_0, v_1, \cdots, v_{n-1}, \underbrace{0, 0, \cdots, 0}_{m}]. \tag{3.13}$$

A mapping Ψ_C is called optimal if the average total power consumption:

$$\hat{E}_t = \sum_i \sum_j p(w^{(i)}, w^{(j)}) \cdot \mathcal{E}[w^{(i)}, w^{(j)}] \tag{3.14}$$

is minimized, where $w^{(i)}$ represents the i-th codeword of the output alphabet while the term $p(w^{(i)}, w^{(j)})$ denotes the probability that a transition from state $w^{(i)}$ to state $w^{(j)}$ occurs on the bus, i.e. at the output of the coder.

The purpose of each coding scheme is to find the optimal Ψ_C or at least construct a mapping as efficient as possible at the expense of the highest affordable complexity. The mapping can imply widening the bus or not. In order to construct such a mapping, the transition probabilities have to be estimated *a priori* at design time or *a posteriori* at runtime.

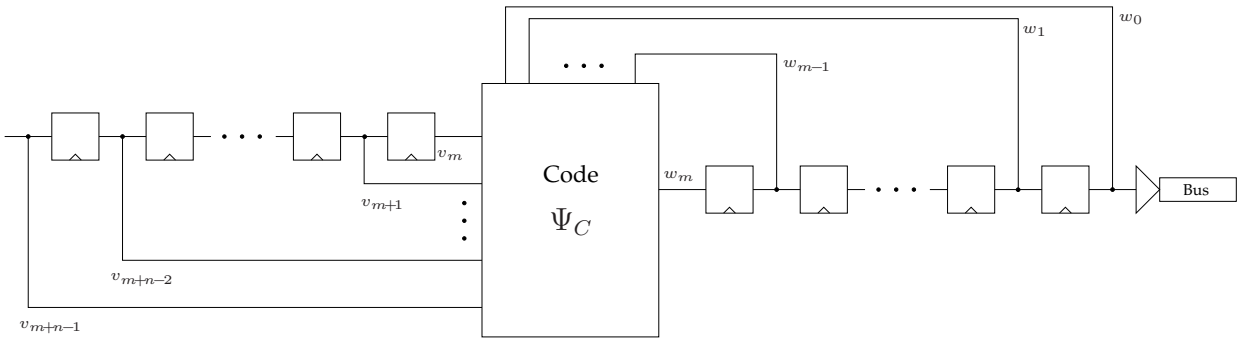


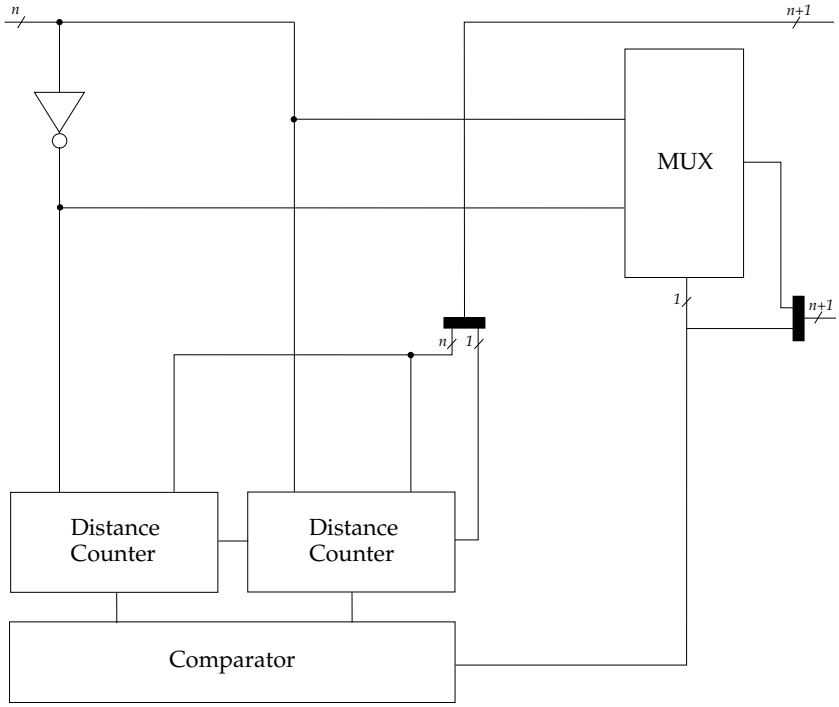
Fig. 3.7: Coding for power. Problem formulation

From the perspective of the knowledge about the encoded data, coding schemes can be divided in two main categories: application-specific schemes that exploit the knowledge about the data and general schemes that are less efficient in some specific cases but have a larger applicability. Coding schemes can be static or adaptive depending on their ability to adjust themselves to the statistical characteristics of the transmitted data.

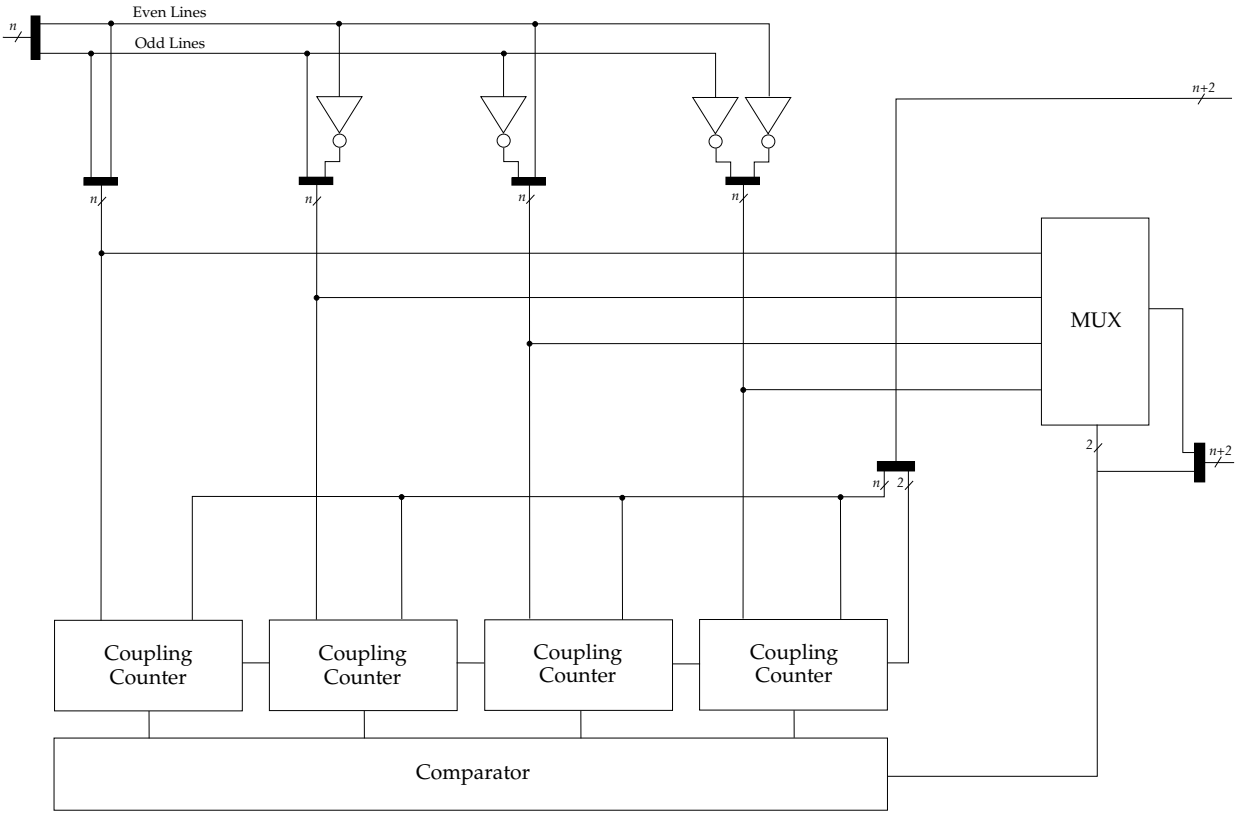
One of the most common and widely applied encodings due to its simplicity and versatility is the bus invert scheme proposed by Stan and Burleson in [187]. Bus invert compares based on the previously sent data whether it is more power efficient to send the data negated or not. As shown in **Fig**. 3.8 a), it adds for this purpose one extra line which signalizes the inverting. In the case of uncorrelated uniformly distributed data, bus invert is optimal among one-bit redundant encoders [187, 189]. Instead of computing the Hamming distance to the previously sent codeword, one can envisage to minimize the transition activity associated to the coupling capacitances by computing the so-called coupling distance as shown in [83]. Throughout this thesis we refer with *Bus Invert Hamming (BIH)* and *Bus Invert Coupling (BIC)* to the classic Hamming-distance based bus invert scheme and the bus invert scheme applied for reducing the coupling activity, respectively.

The coupling activity can be reduced even more by treating odd and even lines differently as in proposed in the so-called Odd/Even Bus Invert (OEBI) scheme [206]. Odd and even lines are inverted separately. Hence, significant reductions in dynamic power consumption are obtained by comparing the coupling activity for the four possible cases: not inverting any line, inverting only the odd lines, inverting only the even lines, and inverting all lines. Obviously, OEBI requires two signalization bits and its implementation is more complex than that of the one-bit redundant bus invert schemes as illustrated in **Fig**. 3.8 b). However, the versatility of OEBI is much higher due to those extra bits.

As previously mentioned, BIH is optimal for uncorrelated uniform data. However, in the case of buses with lines exhibiting a low activity due to high correlation as well as a high activity as a result of poor correlation, BIH is not a good choice. In order to cope with this problem, Shin et al. developed in [171] the so-called Partial Bus Invert (PBI) code. Basically, BIH is applied only to a selected set of poorly correlated bits by using a precomputed mask. The mask can be computed also at run-time if the data statistics are unknown



(a) Bus Invert (after [187])



(b) Odd/Even Bus Invert (after [206])

Fig. 3.8: Bus Invert schemes

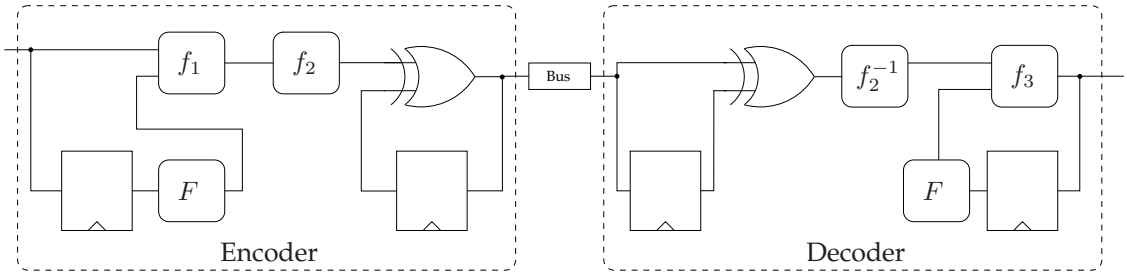


Fig. 3.9: Coding framework for self transition activity reduction (after [143])

at design time and subject to changes during operation [87, 88, 89, 90]. Other interesting and more general schemes are the Limited-Weight Codes (LWC) [185, 186, 188, 189, 190], modified bus invert schemes [63, 126], or (adaptive) dictionary-based schemes [85, 104].

The first adaptive encoder has been introduced by Benini et al. in [15]. The main feature of the code is that it does not possess a fixed mapping function, but calculates on-the-fly the most suitable one based on the extracted data statistics. For the selected data sets, the adaptive scheme reduces the activity about 10% while bus invert achieves an improvement of only around 2.5%.

The efficiency of the so-called Frequent Value Encoding proposed in [202] relies on the requirement that a large amount (58% to 68%) of the data that can be transmitted via a bus has a large probability of appearance. Those symbols are encoded *one-hot*, that is the weight of the resulting codewords is equal one.

Sotiriadis proposed Transition Pattern Coding (TPC) in [177, 178] in order to reduce coupling activity in closely spaced buses. TPC introduces redundancy and increases thus the number of possible codewords at the encoder output and selects a power-efficient mapping based on a heuristic. However, the extreme complexity of the heuristic reduces significantly the applicability of the method.

In application-specific encoding schemes, essential characteristics of the word-level and bit-level activity are exploited in order to reduce power consumption. A very important class of applications which has been widely considered for reducing power are address buses. The most important schemes are Gray-coding [132], Asymptotic Zero-Transition Activity (T0-Code) [13], T0-BI, Dual-T0, Dual-T0-BI [14], Beach Solution [12], Working Zone Encoding [122], Self-Organizing Lists [108], Selective Line Alignment [107], Extended Transition Activity Measure (ETAM++) [95]. Moreover, power consumption of dedicated processors can be efficiently decreased through instruction set encoding [11].

Ramprasad et al. introduced in [143] a coding framework for low-power address and data buses. In the framework represented also in **Fig**. 3.9, a data source (characterized in a probabilistic manner) is first passed through a decorrelating function f_1. Afterwards, a variant of entropy coding function f_2 is employed, which reduces the line transition activity. The framework can be employed to construct and analyze encoding schemes by implementing different functionalities for f_1 and f_2. The function F is a predictor

of the current input values based on past values. For instance, in address buses F is just the incrementing function as addresses tend to be highly correlated. Implementation possibilities for f_1 are XOR, difference-based mapping (dbm), or value-based mapping (vbm), while invert (inv), probability-based mapping (pbm), vbm, or dbm are possible choices for f_2. The main drawback of the framework is its limited applicability since it has been tailored for reducing the line transition activity and not the coupling one.

Due to their usually large data buses and their wide spread, DSP architectures represent a very interesting application domain for constructing efficient low-power bus codes. However, there is still a lack of DSP-specific codes as the only important contributions are transition signaling combined with bus invert (BITS), half-identity half-reverse and transition signaling (hihrTS) [172]. Coding for reduction of power consumption in LCD displays and high-speed video interfaces can be regarded to be loosely related to DSP applications, as the image data exhibit an important correlation [18, 27, 129, 161, 162].

Furthermore, several methods, mainly modifications of bus invert or odd/even bus invert, have been proposed for reducing bus delay, improve throughput, and decrease crosstalk-induced noise [60, 81, 92, 158, 181, 193, 196]. Rao et al. proposed in [147] a bus encoding algorithm and circuit scheme for on-chip buses that simultaneously decreases capacitive crosstalk and leakage-induced power consumption. Inductive noise can be reduced by the so-called bus stuttering method, that inserts dummy states in order to avoid worst-case noise generating states [91].

3.5 Summary

This chapter consisted of a general overview on the state-of-the-art in interconnect optimization. The survey covered optimization techniques at several levels: technological, layout and routing, circuit, and architectural and system level. At the technological level, two fundamental mainstreams exist. On the one hand, manufacturing and device engineers try to improve existing technologies by incorporating new design methods and materials. On the other hand, unconventional and radical interconnect solutions like optical interconnects, carbon nanotubes, cooled superconductors, RF/microwave on-chip communication, or molecular interconnects are under investigation.

At the layout and routing level, the most commonly used techniques are increased metal separation (wire spacing), shielding, wire sizing, wire splitting, and interconnect routing. The first two techniques behave differently depending on the type of crosstalk, i.e. inductive or capacitive. The so-called 45° routing technique is more efficient in terms of required routing resources, however it is much more prone to inductive crosstalk effects. Wire sizing, shaping, and especially splitting can be efficiently used in inductive interconnects.

In order to reduce transmission line effects, line terminations or matching is a set of techniques that have been extensively used in PCB design. The main idea is to add resis-

tive and/or capacitive or diode-based line terminations such that reflections are canceled. Further, buffer insertion is probably still the most popular and widely used delay optimization technique. Buffers can be inserted in RC lines as well as in RLC lines to reduce interconnect delay. In order to improve delay and power in interconnects, advanced signaling techniques can be used. For instance, a reduced line swing can be employed as power consumption depends quadratically on the voltage step.

Since the scope of this thesis is high-level interconnect optimization, the focus of this chapter has been laid more on architectural and system-level techniques. Because interconnect tends to become the main factor – or at least one of the most significant ones – in overall system performance and power consumption, interconnect optimization has to be addressed at high levels of abstraction. In this context, an interconnect-centric design flow is reviewed. Moreover, several promising architectural paradigms like multi-cycle communication, global asynchronous locally synchronous (GALS) communication, and communication-centric platform-based design (NoCs) are briefly discussed. Finally, a multitude of signal encoding schemes that aim at reducing crosstalk, signal delay, and/or power consumption (especially for decreasing the self activity) are outlined. The primary attention is given to bus invert (BI) and odd/even bus invert (OEBI), as these two schemes represent the fundament for the signal encoding techniques developed in this thesis.

Chapter 4

Analysis and Macromodeling of Delay and Power Consumption

Contents

The performance of digital systems is determined by the delay of signals through combinational logic between clocked registers or memory elements. The delay of those paths is decided by both logic gates and increasingly by interconnects that become the dominant factor due to VDSM effects. A multitude of delay estimation methods of different complexity have been proposed based for instance on moment calculation or worst case alignment. However, as mentioned in [30], in interconnect synthesis and optimization it is extremely important to keep in mind system-level design goals. Therefore, the right balance between accuracy and efficiency must be kept.

Generally, the Elmore delay predictor was able to assure that objective as it rapidly provided for most interconnect structures sufficiently exact estimations. Nonetheless, the Elmore model can be highly inaccurate in VDSM interconnects. For instance, it cannot handle resistive shielding, slew dependency, or inductive effects. Therefore, higher order delay metrics are required. Furthermore, when coupling effects are not negligible anymore, the performance is determined by the worst case. In order to estimate those worst cases, the existing delay models have to be enhanced or replaced by new ones. On the contrary, power consumption is not defined by worst cases, but by the weighted average of the effect of all possible patterns.

If all patterns are allowed and if the system level objective is to estimate the largest existing delay, one can employ worst case models. However, if the goal is to improve performance by coding, i.e. eliminating a sufficient set of worst cases, new models able to predict the delay for all or at a least a significant amount of input patterns are required. Therefore, the objective of this chapter is to construct a pattern-dependent delay model in buses exhibiting capacitive and inductive coupling and to extract out of accurate power macromodels that take into account also coupling effects the core information necessary for high-level optimization of performance and power consumption based on signal encoding.

This chapter is organized as follows. First, several delay models are briefly discussed like Elmore, moments-based, and worst case. Afterwards, the core of this chapter is represented by the construction of a pattern-dependent delay model in capacitively and inductively coupled buses. Lastly, the employed macromodel for power consumption is presented.

4.1 Delay Models

Once a gate and interconnect delay issue is translated into a linear RC problem, several delay estimation methods can be applied. Due to its intrinsic simplicity, the Elmore delay model introduced in [42] has been the most widely used. However, the Elmore delay model is inaccurate in VDSM interconnects, as it cannot cope with several important effects like resistive shielding, slew dependency, or inductive effects [26]. The limitations of the Elmore model can be overcome by employing higher order delay metrics. Moreover, when capactive coupling appears, the worst-case delay can also be estimated [164].

4.1.1 Elmore Delay

As technology shrinks and on-chip metal wires tend to have a significant resistance, the equipotential assumption of the lumped-capacitor model does not hold. Therefore, a more adequate RC model had to be introduced. The Elmore delay has been initially developed for estimating the delay of wideband amplifiers [42], but Rubinstein et al. ob-

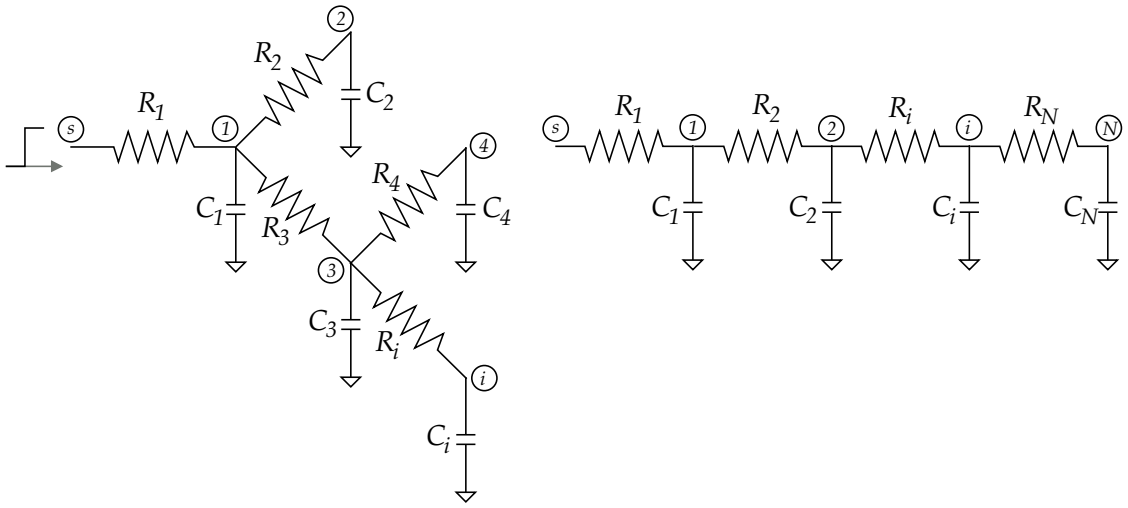


Fig. 4.1: Tree-structured RC network and RC chain (after [141])

served later its usefulness in estimating delay of RC trees [154]. Further, lumped RC model are suitable only if the distributed nature of lines can be disregarded. Otherwise, network RC models approximating the distributed behavior of interconnects are required as illustrated in **Fig**. 4.1 for an RC tree and an RC ladder or chain. However, solving analytically large networks for delay is not a trivial task and Spice simulations are also very complex and time-consuming. Therefore, simple delay estimators like the Elmore one are necessary.

In the abovementioned topologies there exists a unique resistive path between a source node s and any node i of the network [141]. For instance, the path resistance between source node s and node i is defined as $R_{ii} = R_1 + R_3 + R_i$. By defining the so-called shared path resistance between nodes i and j R_{ij}, i.e. the resistance shared among the path from s to i and the one from s to j, the Elmore delay, τ_{Di}, for a node i can be calculated as:

$$\tau_{Di} = \sum_{j=1}^{N} C_j R_{ij} \tag{4.1}$$

where N represents the total number of network nodes. In the case of ladder network, the Elmore delay becomes:

$$\tau_{DN} = \sum_{j=1}^{N} C_j R_{jj}. \tag{4.2}$$

In a symmetric RC ladder network, $R_i = R_j = R$ and $C_i = C_j = C$, $(\forall) i, j = \overline{1, N}$. Thus:

$$\tau_{DN} = \frac{N+1}{2N} RC \xrightarrow[N\to\infty]{} \frac{RC}{2}, \tag{4.3}$$

which represents half of the dominant time constant predicted by the more pessimistic lumped model. The Elmore delay is defined by the first moment of the impulse re-

sponse [164], and the 50% delay in node i, t_{di}, is thus given by:

$$t_{di} = \ln 2 \cdot \tau_{Di} \simeq 0.69 \tau_{Di} = 0.69 \sum_{j=1}^{N} C_j R_{ij}. \tag{4.4}$$

As shown also in **Chap**. 7, the main advantage of the Elmore delay comes from the fact that it is recursively decomposable and therefore, it can be easily employed in a set of effective algorithms. Therefore, by enhancing the Elmore delay with look-up table estimators, industrial design engineers partly managed to stick to Elmore-based interconnect synthesis tools [30].

In order to cope with the increasing interconnect self-inductance, Ismail and Friedman developed in [70, 71] closed form solutions for delay, rise time, overshoots and settling times in RLC trees. The delay estimator has the same accuracy as the Elmore delay for RC trees, and more importantly, it preserves the computational simplicity and the recursive properties. It is shown that the delay in node i of an RLC tree network is:

$$t_{di} = \frac{1.39\,\xi_i + 1.047\,\mathrm{e}^{-\frac{\xi_i}{0.85}}}{\omega_{ni}}, \tag{4.5}$$

where

$$\xi_i = \frac{1}{2} \frac{\sum_{j=1}^{N} C_j R_{ij}}{\sqrt{\sum_{j=1}^{N} C_j L_{ij}}}, \tag{4.6}$$

$$\omega_{ni} = \frac{1}{\sqrt{\sum_{j=1}^{N} C_j L_{ij}}}. \tag{4.7}$$

Note that t_{di} can be also written as:

$$t_{di} = 0.69 \sum_{j=1}^{N} C_j R_{ij} + \frac{1.047 \mathrm{e}^{-\frac{\xi_i}{0.85}}}{\omega_{ni}} = 0.69 \tau_{Di} + \frac{1.047 \mathrm{e}^{-\frac{\xi_i}{0.85}}}{\omega_{ni}} \tag{4.8}$$

which shows that in the case of a low self-inductance (small values of ξ_i and ω_{ni}), the delay estimator is that of the RC case.

4.1.2 Moments-based Delay Metrics

The Elmore delay metric is actually the simplest and most effective first order expression of the broader idea of model order reduction in which a higher order system is approximated by an equivalent lower order model that is able to capture the essential characteristics of the original system [164]. In [139], Pillage and Rohrer introduced the so-called

Asymptotic Waveform Evaluation (AWE), a methodology that provides the possibility to perform such an order reduction based on the moments of the transfer function of the system. The fundamental idea of the AWE is to calculate the circuit moments and employ a moment matching method to compute Padé approximations of the circuit transfer function.

In the case of a single-input single-output linear system with transfer function $h(t)$, the k-th moment m_k is defined as:

$$m_k = \frac{(-1)^k}{k!} \cdot \int_0^\infty t^k h(t) dt. \tag{4.9}$$

It can be easily shown (see [139,164]) that by approximating the Laplace transform of $h(t)$, $H(s)$, with a MacLaurin series expansion, the following expression is obtained:

$$H(s) = \sum_{k=0}^{\infty} m_k s^k. \tag{4.10}$$

Once the moments of the transfer function are computed, the Padé approximation can be applied for moment matching. The accuracy of the transfer function approximation and implicitly that of the delay estimator depends on the number of employed moments.

Based on the first three or only first two moments, several empirical metrics for the 50% delay in RC networks have been determined [139]:

$$t_d = \tau_D \cdot \overline{m}_2^2 \cdot \frac{\overline{m}_3 b_1 - b_2 + \dfrac{b_3}{\overline{m}_3}}{\overline{m}_3 a_1 - a_2 + \dfrac{a_3}{\overline{m}_3}} \tag{4.11}$$

$$t_d = \frac{1}{2} \cdot \Big(- m_1 + \sqrt{4m_2 - 3m_1^2} \Big) \cdot \ln \left(1 - \frac{m_1}{\sqrt{4m_2 - 3m_1^2}} \right) \tag{4.12}$$

$$t_d = \sqrt{2m_2 - m_1^2} \cdot \ln 2 = 0.69 \tau_D \cdot \sqrt{\frac{2m_2}{\tau_D^2} - 1} \tag{4.13}$$

$$t_d = \frac{m_1^2}{\sqrt{m_2}} \cdot \ln 2 = 0.69 \tau_D \cdot \frac{\tau_D}{\sqrt{m_2}} \quad \text{(the so-called D2M metric)} \tag{4.14}$$

where the coefficients a_i and b_i are calculated via least squares approximations using two-pole approximations, and $\overline{m}_k$ is the normalized moment of order k, i.e. scaled by the Elmore delay ($-m_1$):

$$\overline{m}_k = (-1)^k \cdot \frac{m_k}{m_1}. \tag{4.15}$$

Furthermore, several methods employ the probabilistic interpretation of moments to construct a delay metric among which the most commonly known are: the h-gamma

metric that is based on splitting the Laplace transform of the step response in a forced response and a homogeneous one, the PRIMO (Probability Interpretation of Moments for Delay Calculation) metric in which the impulse response is fitted to the gamma distribution, and the WED (Weibull-based Delay) metric that utilizes a Weibull distribution instead of the gamma distribution [164].

4.2 Pattern-Dependent Delay Modeling

The maximum operating frequency is determined by the longest path a signal has to travel between sending and receiving units. When the delay of a line does not depend on the transition occurring in other lines, the problem of finding the longest path through an interconnect structure can be reduced to calculating one of the aforementioned metrics or using a simple table-based approach. However, in coupled interconnects, the togglings on the neighboring line (aggressors) significantly affect the delay on the analyzed lines (victims). Therefore, the problem of determining the worst case delay is equivalent to finding the input assignment or pattern that produces that worst case delay.

The abovementioned delay metrics for RC networks cannot be applied in coupled RC networks or underdamped RLC networks. For non-coupled underdamped RLC interconnects, it has been shown that central moments can be used to analyze the resulting waveform and predict phase delay.

In the case of capacitive coupling, crosstalk can be easily modeled by replacing the coupling capacitor with a so-called Miller capacitance [164]. The advantage of the simple Miller capacitance is that the coupled interconnect is translated into a set of uncoupled systems, each of those being solved employing the same methods previously described. Nevertheless, the exact value of an equivalent Miller capacitor depends on the moment and direction of the toggling in the aggressors. So, the aggressor alignment is decisive in finding the worst case delay scenario [16, 164].

4.2.1 A Linear Delay Model for Capacitively Coupled Buses

In a realistic chip situation that comprises an impeding amount of aggressor-victim scenarios, it is of utmost importance to drastically reduce the cases that have to be taken into consideration to an acceptable number. Such a simplification can be conducted especially because the influence of each aggressor is of a different type and magnitude due to numerous factors.

As an effect of technology scaling, the coupling capacitance between neighboring wires increased with each technology node and is currently dominating the overall line capacitance. Nevertheless, the second-order coupling capacitances are almost perfectly shielded by the first-order ones. Thus, the capacitive coupling of one bus line with the two neighbors determines the dynamic power consumption and the time required for

a transition to complete in that bus line. The transition time is determined by the relative transitions of the lines. For example, a victim toggling from low to high has to charge twice the coupling capacitance to a neighbor switching from high to low because of the Miller effect. On the contrary, when the neighbor switches in the same direction, the coupling capacitance does not have to be charged. This way, the delay function of capacitively coupled lines can be easily constructed as shown in detail in [175, 180].

In the sequel, we denote the transition in line i in an n-bit wide bus as $\Delta b_i = b_i^+ - b_i^-$, where b_i^- and b_i^+ represent the initial and final value on line i, respectively. We also define the transition vector, $\underline{\Delta b} = [b_1, b_2, \cdots, b_n]^t$. Basically, each line is characterized by the effect on the delay produced by four possible switching scenarios in each aggressor, namely $(b_i^-, b_i^+) = \{(0,0), (0,1), (1,0), (1,1)\}$. Generally speaking, the set of line delays is a function of the transition vector.

In capacitively-coupled interconnects, when inductances can be safely ignored, the delay in line k, δ_k, of a symmetric bus is given in [175] as:

$$\delta_k = \tau_0 \big[(1+2\kappa)\Delta b_k^2 - \kappa \Delta b_k (\Delta b_{k-1} + \Delta b_{k+1})\big], \tag{4.16}$$

where τ_0 is the delay of the crosstalk-free line and $\kappa = \frac{C_c}{C_s}$ is the ratio of the coupling capacitance C_c and the ground capacitance C_s, the so-called *bus aspect factor*[1]. In a bus, due to the shielding effect of the first-order neighbors on the higher-order ones in terms of coupling capacitance, the effect of the aggressors of an order higher than two can be neglected without any accuracy loss. Note that for convenience the term Δb_k^2 is used instead of $|\Delta b_k|$.

4.2.2 An Extended Linear Delay Model

In order to address timing issues at higher levels of abstraction, accurate models capable to predict pattern dependent signal delay are required. This analysis is mandatory if delay-aware coding is to be employed in high-speed VDSM buses. Current techniques restricted only to non-inductively coupled interconnects because of a lack of proper compact models [175, 181]. Some efforts have been put in identifying worst-case switching patterns in inductively coupled lines [192]. However, they cannot predict the delay for a given input switching pattern. Further, the delay coding limits and delay improvement methods developed in [175] for capacitive coupling do not hold in the more general case of inductively-coupled lines and have to be revised.

The goal is to develop a high-level model which predicts the delay in dedicated point-to-point interconnects for all switching patterns. The technique has to take into account the pattern-dependent behavior of the delay as well as the effect of process variations on delay. The developed delay model includes inductive coupling effects and is basically a generalization of the one proposed in [175] for buses with inter-wire capacitance and negligible inductive effects.

[1] C_g and C_s are used interchangeably to denote the ground (self) capacitance

In the sequel, a 5-bit wide bus structure is considered as an example. Width, w, thickness, t, and pitch p are chosen as 1 µm, 1 µm, and 3 µm, respectively. The distance to the lower and upper metal layer is considered to be 1 µm. To obtain accurate interconnect parameter values, the wires were split into segments of less than 100 µm length. The number of segments, s, has been increased along with the wire length to maintain sufficient accuracy. Both isolated and non-isolated buses are considered. The bus margins are considered to be ground wires that have the same geometry as the signal lines. For calculating the intrinsic delay of the wire, the methodology described in **Chap**. 2 has been employed. For the selected simulation sets, we have noticed that we can use an equivalent series impedance of 75 Ω and only to slightly change the trapezoidal input pulse for a good approximation of the line input signal. Further, the far-end buffers have been modeled with equivalent capacitances (50 fF).

Eq. (4.16) can be rewritten in a more general fashion:

$$\delta_k = \alpha_k \Delta b_k^2 + (\alpha_{k-1}\Delta b_{k-1} + \alpha_{k+1}\Delta b_{k+1})\Delta b_k \tag{4.17}$$

$$= \sum_{i=k-1}^{k+1} \alpha_i \Delta b_i \cdot \Delta b_k, \tag{4.18}$$

where $\alpha_k = \tau_0(1 + 2\kappa)$, $\alpha_{k-1} = \alpha_{k+1} = -\tau_0\kappa$. It is worth mentioning here, that for non-inductive interconnects, α_k is positive, while α_{k-1} and α_{k+1} are negative, and that in the case of non-symmetric buses we generally have $\alpha_{k-1} \neq \alpha_{k+1}$.

Inductive coupling is a long-range effect in contrast to the short-range capacitive coupling. Therefore, the effect of the aggressors of order higher than two cannot be discarded for an accurate analysis. The abovementioned linear pattern-dependent delay model can be extended in order to include inductive coupling between neighbors of order higher than two. It can be seen in the sequel, that the signal delay can be approximated as a linear combination of the delay produced by the switching patterns on every line.

A simple and efficient approach to approximating the impact of capacitive coupling is to include its effect into the equivalent capacitance seen by the gate, either by adding a term to the ground capacitance or by multiplying it with a Miller factor [26,164,175]. The added term is pattern-dependent as the effectively seen coupling capacitance depends on the relative togglings on the victim and the aggressors.

Conceptually, the problem can also be formulated by choosing a nominal pattern and constructing an equivalent pattern-dependent interconnect model instead of computing equivalent effective capacitances, inductances, or resistances. For this purpose, a nominal pattern is selected for each line, for instance the one when the victim line toggles from low to high and all aggressors are quiet. Afterwards, for a different switching pattern, a delay matching operation is performed. This means actually, that an equivalent interconnect model with different (pattern-dependent) PUL parameters for each line is constructed, such that the delay of the equivalent network under the nominal toggling pattern is equal to the delay of the real interconnect model with the actual switching pattern as input.

Consider the example with the influence of coupling capacitances on delay as a function of the switching pattern. As previously mentioned, the additionally seen capacitance induced by the Miller effect can be added to the ground capacitance. This added extra capacitance is a linear function of the relative toggling patterns. Therefore, it can be written in the general case, that the equivalent PUL capacitance of line *k*, C_k, is a linear function of the pattern:

$$C_k(\underline{\Delta b}) = C_{k0} + \sum_{i=1}^{n} \Delta C_i \Delta b_i = C_{k0} + \Delta C_k(\underline{\Delta b}), \tag{4.19}$$

where C_{k0}, the PUL capacitance for the nominal pattern, is in general a non-linear function of many parameters like rise time, load impedance, and technological parameters, while ΔC_i is the extra capacitance due to the coupling to line i.

For simplicity, we can assume that for finding every equivalent PUL parameter, we have to add a linear term in Δb_i. However, in the case of inductances (as well as for several other parameters), this assumption introduces higher errors.

Let ψ_j be a PUL parameter or any other parameter one has to compute for delay matching and let m be the number of such parameters. The set of all those parameters are defined as $\underline{\Psi} = \{\psi_1, \psi_2, \ldots \psi_m\}$. As a result of the abovementioned assumption, we can write for the delay matching:

$$\psi_j(\underline{\Delta b}) = \psi_{j0} + \sum_{i=1}^{n} \Delta\psi_{ji} \Delta b_i = \psi_{j0} + \Delta\psi_j(\underline{\Delta b}), \tag{4.20}$$

where ψ_{j0} represents a (non-linear) function of many factors, and $\Delta\psi_{ji}$ denotes the difference in ψ_j in line i. In particular, we have $\underline{\Psi} = \{R, L, M, C\}$.

The delay of a line can be expressed for a given bus and driver configuration as a continuous function of the PUL parameters. By neglecting the non-linear terms in Δb_i of the Taylor expansion around $\underline{\Psi_0} = \{\psi_{j0}\}_{j=\overline{1,m}}$, we obtain for a low-to-high transition, that is $\Delta b_k = 1$, the following:

$$\begin{aligned}
\delta_k(\underline{\Delta b}) &= \delta_k\big(\underline{\Psi_0} + \underline{\Delta\Psi}(\underline{\Delta b})\big) \\
&\approx \delta_k(\underline{\Psi_0}) + \sum_{j=1}^{m} \frac{\partial \delta_k}{\partial \psi_j} \Delta\psi_j(\underline{\Delta b}) \\
&\approx \delta_k(\underline{\Psi_0}) + \sum_{i=1}^{n} \left(\sum_{j=1}^{m} \frac{\partial \delta_k}{\partial \psi_j} \Delta\psi_{ji} \right) \Delta b_i.
\end{aligned} \tag{4.21}$$

Hence, when the non-linear terms of the Taylor expansion are negligible, the delay in line k can be expressed as a linear function of the transition patterns in neighboring lines. This observation allows us to extend the delay model of Sotiriadis [175] for inductively coupled lines.

When an aggressor line does not toggle, the effect on delay in the victim line is practically independent of the state. Thus, the contribution of all the patterns $(0, 0)$ and $(1, 1)$

can be modeled by a constant term and the only contributors which must be precisely modeled are the patterns $(0,1)$ and $(1,0)$.

The currents generated by switchings with opposite transitions have opposite directions with opposite transitions. Therefore, the contributions of the patterns $(0,1)$ and $(1,0)$ are of opposite sign though equal as absolute values. The delay predicted in line k is thus:

$$\delta_k = \alpha_k \Delta b_k^2 + \sum_{i=1, i\neq k}^{n} \alpha_{ik} \Delta b_i \cdot \Delta b_k \tag{4.22}$$

$$= \sum_{i=1}^{n} \alpha_{ik} \Delta b_i \cdot \Delta b_k, \tag{4.23}$$

where $\alpha_k \stackrel{\text{def}}{=} \alpha_{kk}$ represents the delay in line k with quiet aggressors, and α_{ik} for $i \neq k$ denotes the contribution to the delay of the aggressor line i on line k. We call this model the *Extended Linear Delay (ELD) Model* and the corresponding α_{ij}-s *model coefficients* or simply *coefficients*.

The ELD model can be written in a compact way also for an n-bit wide bus. For this purpose, we consider the following notations:

$$\underline{\delta} = [\delta_1, \delta_2, \ldots \delta_n]^t \tag{4.24}$$

$$\underline{\alpha} = [\alpha_1, \alpha_2, \ldots \alpha_n]^t \tag{4.25}$$

$$\mathbf{A} = [\alpha_{ij}]_{n\times n} \tag{4.26}$$

$$\mathbf{A_i} = diag(\alpha_i) \tag{4.27}$$

$$\mathbf{B} = diag(\Delta b_i) \tag{4.28}$$

It can be easily shown that $\underline{\alpha} = A_i^t |\underline{\Delta b}|$. Thus, the two following forms can be used for the matrix formulation of the ELD model:

$$\underline{\delta} = B \cdot A^t \cdot \underline{\Delta b} \tag{4.29}$$

$$= \underline{\alpha} + B \cdot (A - A_i)^t \cdot \underline{\Delta b}. \tag{4.30}$$

Extensive simulations have been carried out in order to assess the accuracy of the extended linear delay model. Both inductively and capacitively dominated crosstalk scenarios have been covered. The signal delay for a low-to-high transition has been measured as the delay from the time the near-end first reaches 50% of the final value to the time the far-end is stable above 50% of the final value.

As an initial approach, we have calculated the model coefficients on a given line, by performing for a switching on this line SPICE simulations for all possible patterns on the neighbors. The obtained delays are then used to compute the coefficients by the minimum square error approach. It is important to notice, that these coefficients can be precalculated and than used for fast delay estimation.

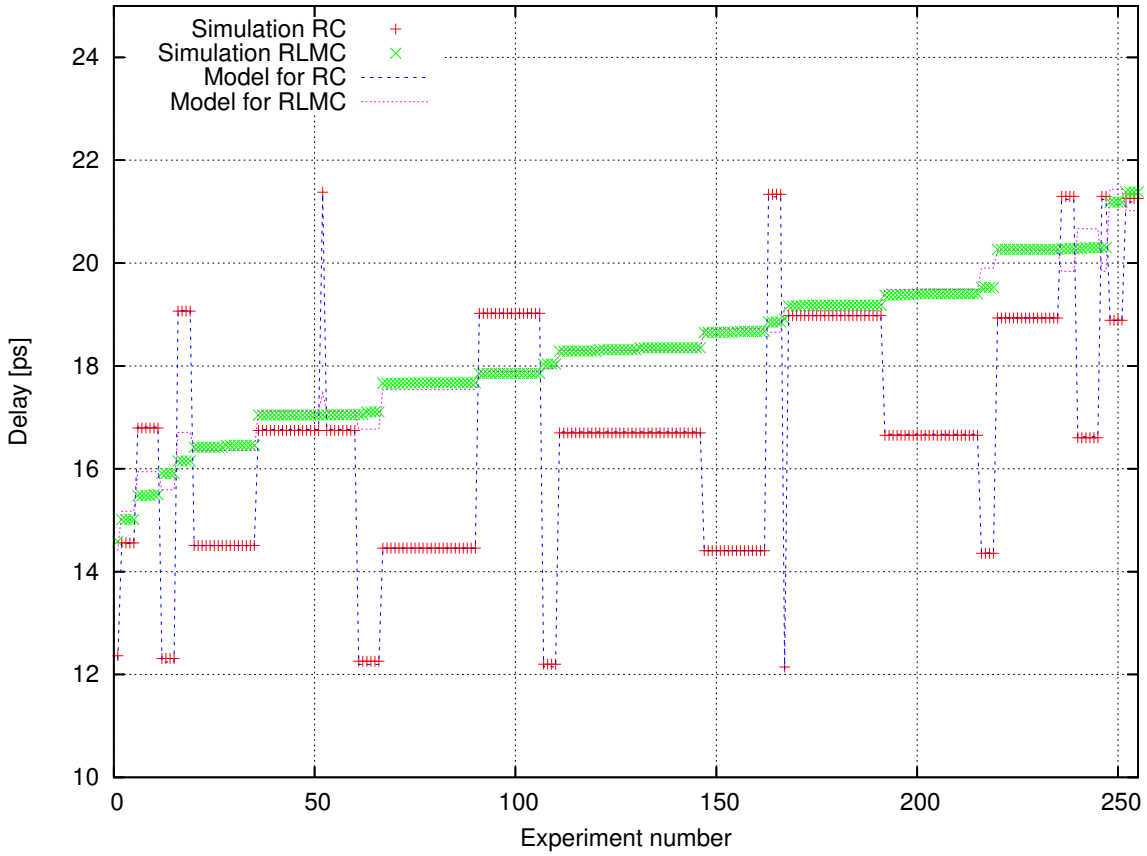


Fig. 4.2: Simulated and estimated delays in RC and $RLMC$ buses

In the case of capacitively dominated coupling, a transition in an aggressor in the opposite direction increases the total capacitance that the victim has to charge, and the transition is thus slowed down, i.e. $\alpha_i < 0$. On the contrary, due to Faraday's law of induction, in purely inductively dominated lines, a transition of an aggressor in the same direction induces a current flowing in the opposite direction to the one in the victim line. Consequently, the effective current decreases and the delay increases, i.e. $\alpha_i > 0$. In buses exhibiting both capacitive and inductive coupling the coefficients for the first-order neighbor can be either negative or positive while the second-order coefficients are negative or very close to zero.

Tab. 4.1: Comparison of ELD and worst case models

	ELD		*WC*	
	ε_{max}	ε_{rms}	ε_{max}	ε_{rms}
RC	2.06	0.78	50.18	28.96
RLC	2.74	0.97	50.77	29.29
$RLMC$	5.34	1.45	58.86	28.56

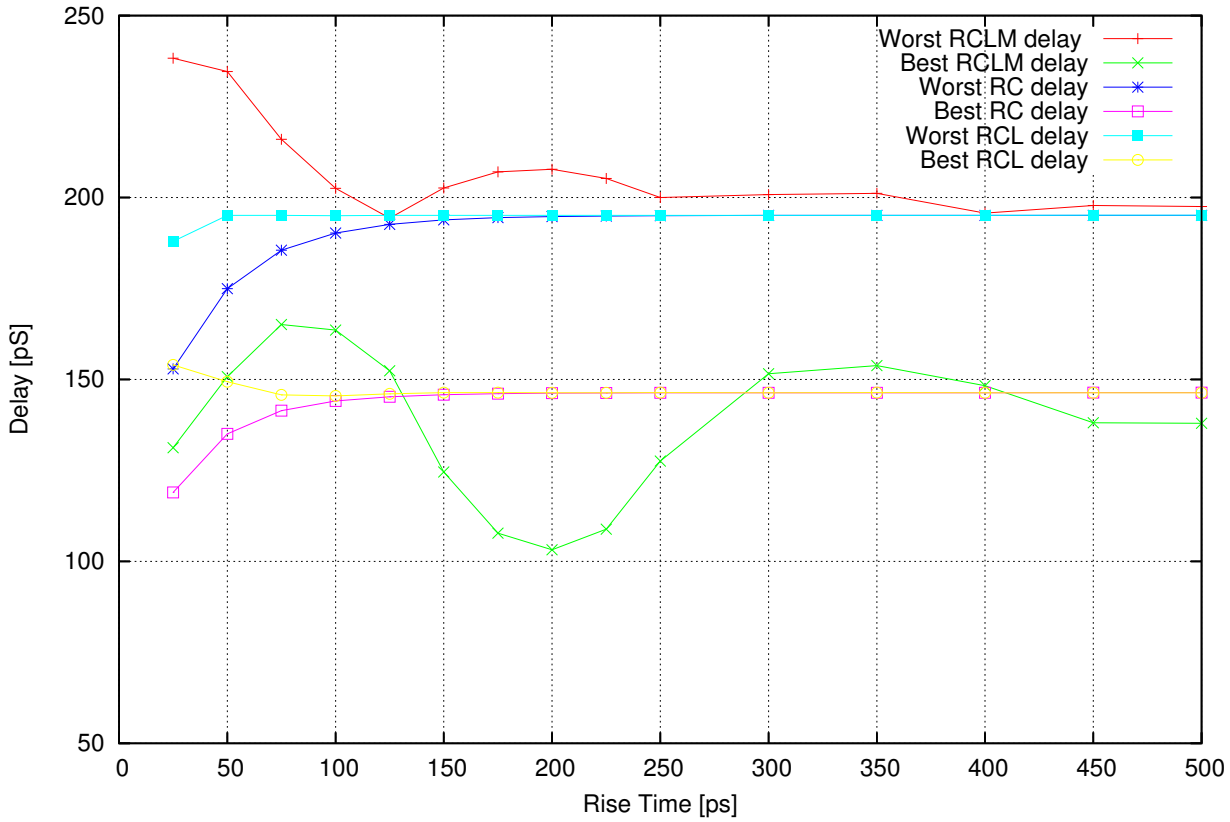


Fig. 4.3: Variation of worst and best case with rise time for the third line of a 1000 µm bus

Tab. 4.1 shows the maximum absolute error, ε_{max}, and the root mean square error, ε_{rms}, of the proposed model for a highly inductive bus. It is to be noticed that previous work taking into consideration only the worst case (WC) introduces very high errors. In general, with regard to capacitive coupling, neighbors of at least second-order are almost completely shielded by the intermediate wires. The short-range nature of capacitive coupling explains the better approximation through the linear assumption. The very small errors appear, for example, because a second-order aggressor may influence very slightly the victim through the two coupling capacitances which separates it from the victim – nevertheless, this influence is extremely small. Further, the assumption of linearity introduces slightly higher errors in the case of inductively coupled interconnects. It is however important to notice, that the chosen case represents a highly inductively coupled line and the error is still small. In the case of the RLC model, the linearity is only marginally affected because line inductances increase the effect of capacitive coupling.

As a typical example, **Fig**. 4.2 shows the simulated and estimated delays for a 1000 µm long line. The rise time has been set to 100 ps. Both RC and $RLMC$ models are depicted for the 256 different switching patterns of the neighbors (the experiments are ordered after increasing values of the $RLMC$ delay). As we can observe, the proposed model fits very well the experimental results. The error for the RC case is almost negligible. For the $RLMC$ network, the error is slightly higher but less than 2 %. Moreover, we can observe

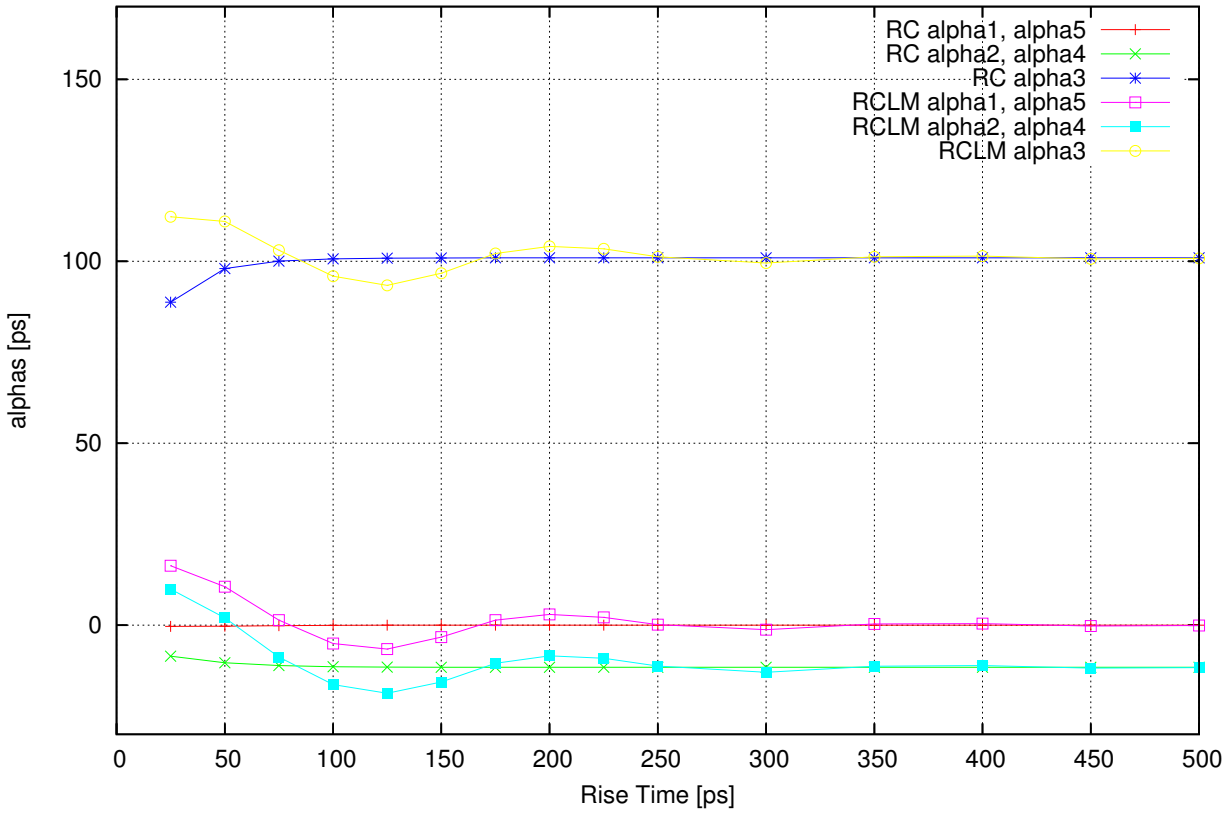


Fig. 4.4: Model coefficients of line 3 for varying rise time for a 500 µm bus

that the worst delay patterns for the RC and $RLMC$ models are completely different and that the simplified model for RC buses is unable to predict the delays induced by the switching patterns in $RLMC$ buses. The proposed ELD model is able to predict with high accuracy that behavior.

The worst and best case delays and switching patterns are almost not changing in RC and RLC interconnects, as shown in **Fig**. 4.3. However, this is not the case for $RLMC$ networks as not only worst and best case delays rapidly change, but also the patterns which induce those delays vary. Moreover, as the rise time decreases, the coupling coefficients change from negative values to positive ones. Thus, the model accurately captures the tendency of the inductive coupling to dominate over the capacitive one with decreasing t_r.

In **Fig**. 4.4, it can be observed that in the case of $RLMC$-interconnects an oscillation of the coefficients with varying rise time. For high rise times, the interconnect behaves capacitively ($\alpha < 0$ for the aggressors) and as the rise time decreases, the interconnect behavior becomes dominated by the inductive coupling components, the coefficients of the neighboring lines becoming thus positive.

Moreover, the delay for all patterns does not increase monotonically with increasing rise times of the input signals as it was the case in simple RC networks. This effect is

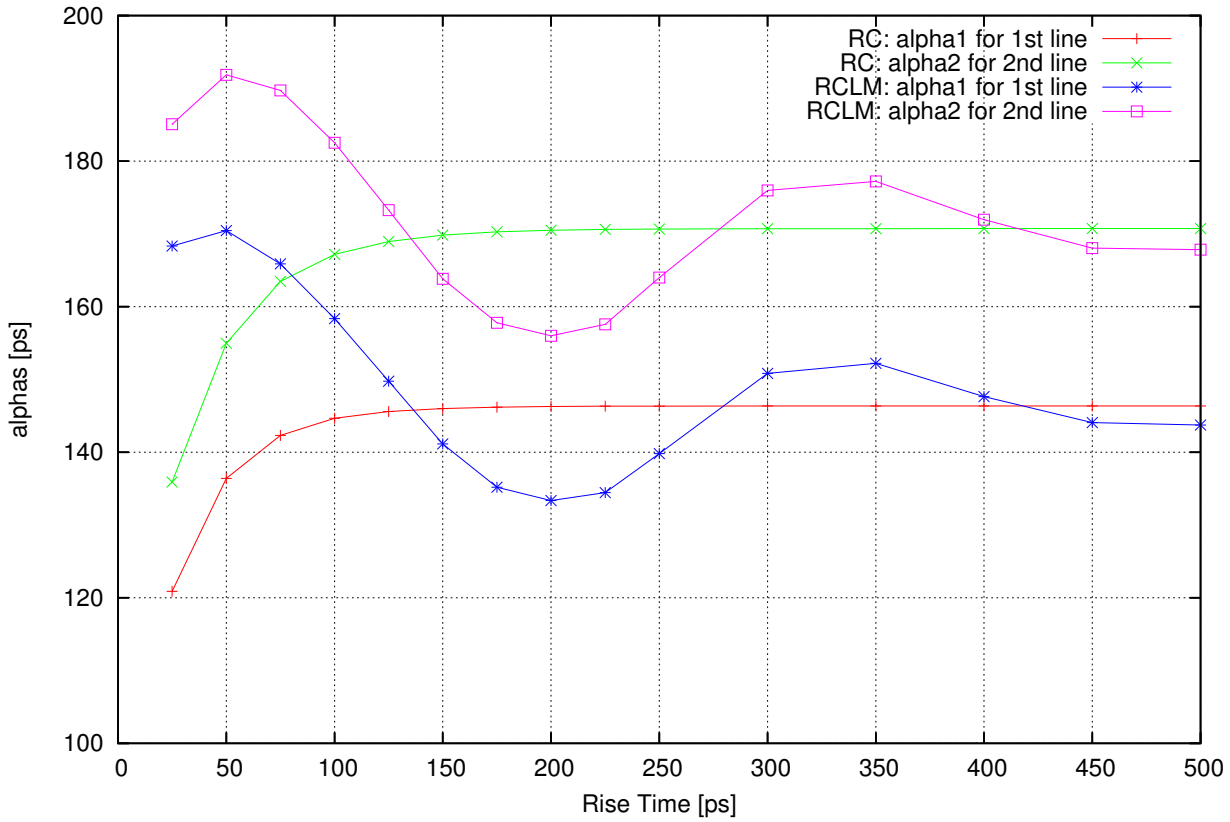


Fig. 4.5: Model coefficients for extreme and middle lines in a 5-bit wide bus

related with the multiple zeros and poles which appear in complex $RLMC$ systems and it is modeled with an oscillating variation of the coefficients with t_r, as we can see in both **Fig**. 4.3 and **Fig**. 4.4.

Concerning the values of the coefficients for different lines, **Fig**. 4.5 shows that α_{11} is clearly smaller than α_{22} for both RC and $RLMC$ interconnect models. The reason behind that is the fact that the first and fifth lines have each only one first-order aggressor (the margins are quite), and not two as in the case of the other three lines. Thus, considerable less crosstalk can be seen in the two extreme lines. Furthermore, in the case of the lines with the same number of neighbors, the coefficients are almost the same. This property of the model coefficients can be used for improving even more the compactness of the model.

In order to show how the model coefficients vary with different parameters, a very large amount of simulations has been performed. **Fig**. 4.6–**Fig**. 4.13 show the variations of α_{23} and α_{24} in an RC and an $RLMC$ modeled bus as a function of the rise time, bus length, and wire pitch. The abovementioned conclusions can be thus understood at a larger scale.

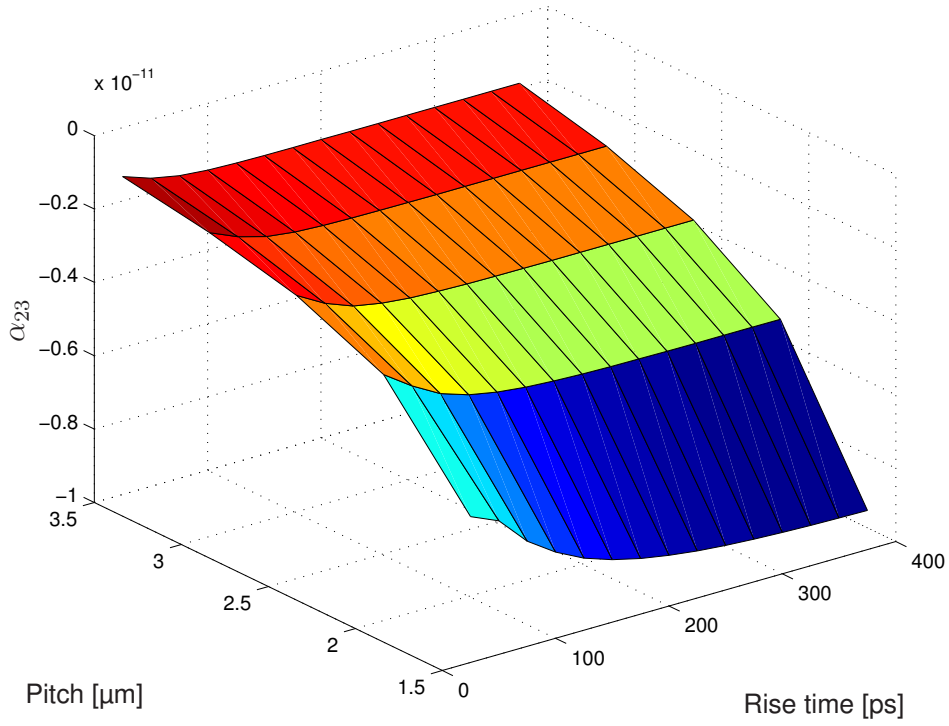


Fig. 4.6: α_{23} as a function of t_r and p in an RC-modeled 5-bit wide bus

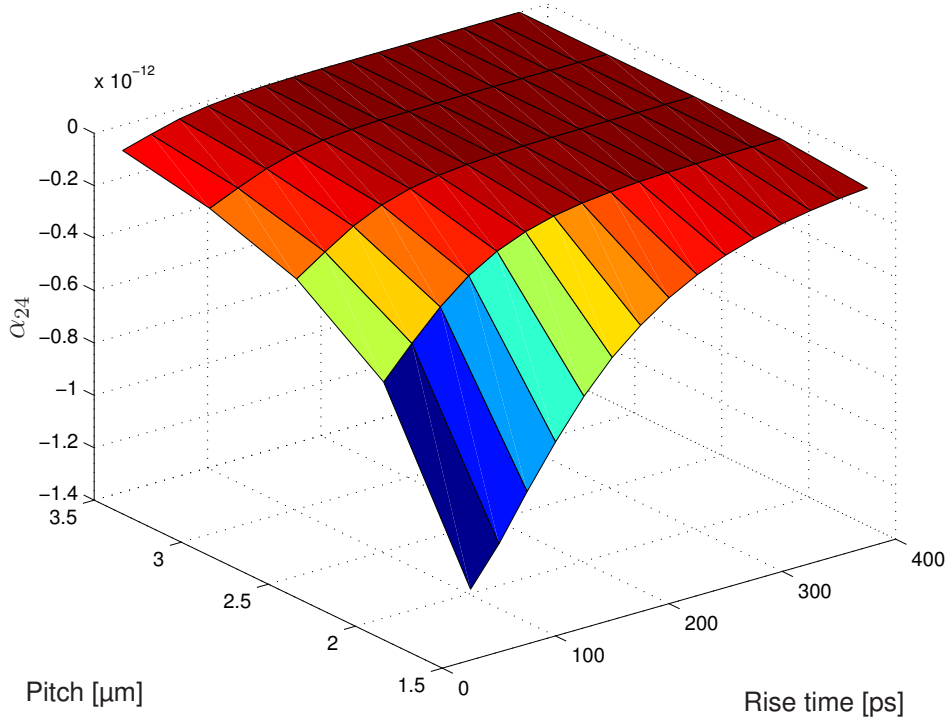


Fig. 4.7: α_{24} as a function of t_r and p in an RC-modeled 5-bit wide bus

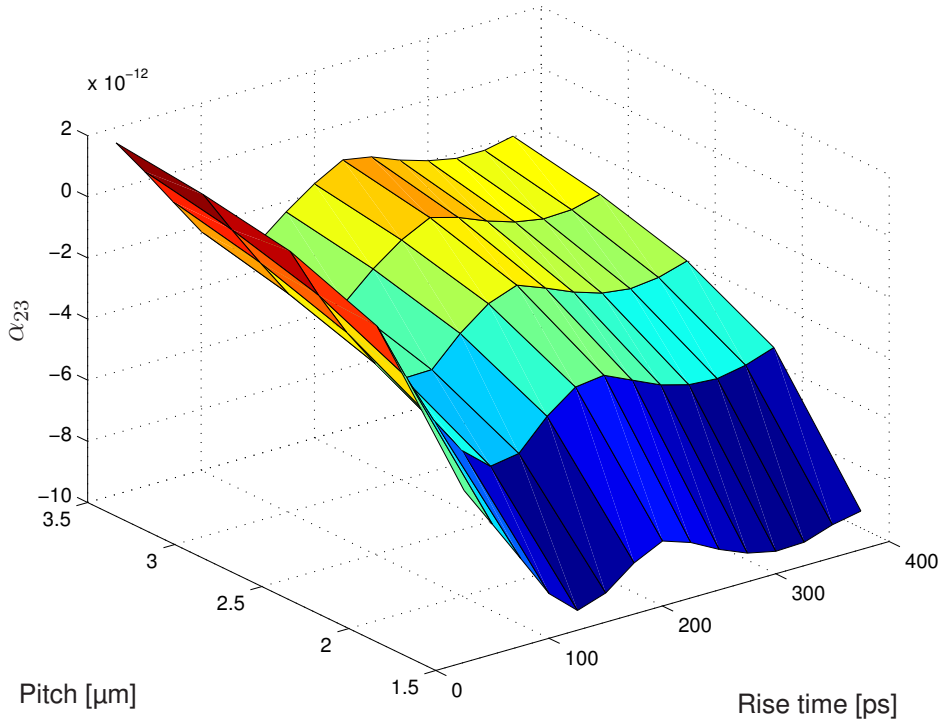


Fig. 4.8: α_{23} as a function of t_r and p in an $RLMC$-modeled 5-bit wide bus

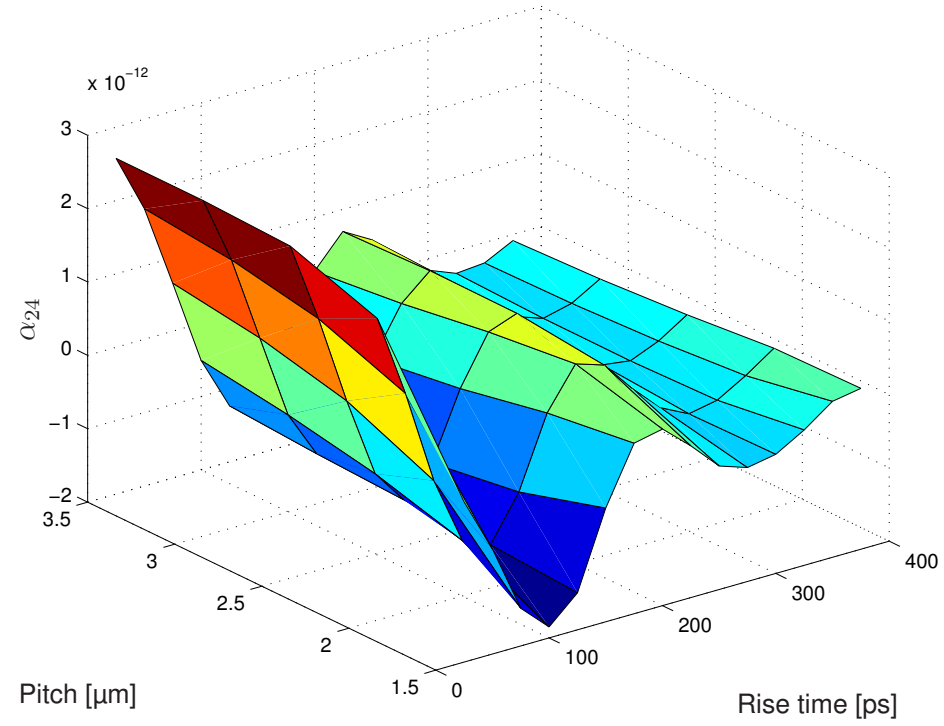


Fig. 4.9: α_{24} as a function of t_r and p in an $RLMC$-modeled 5-bit wide bus

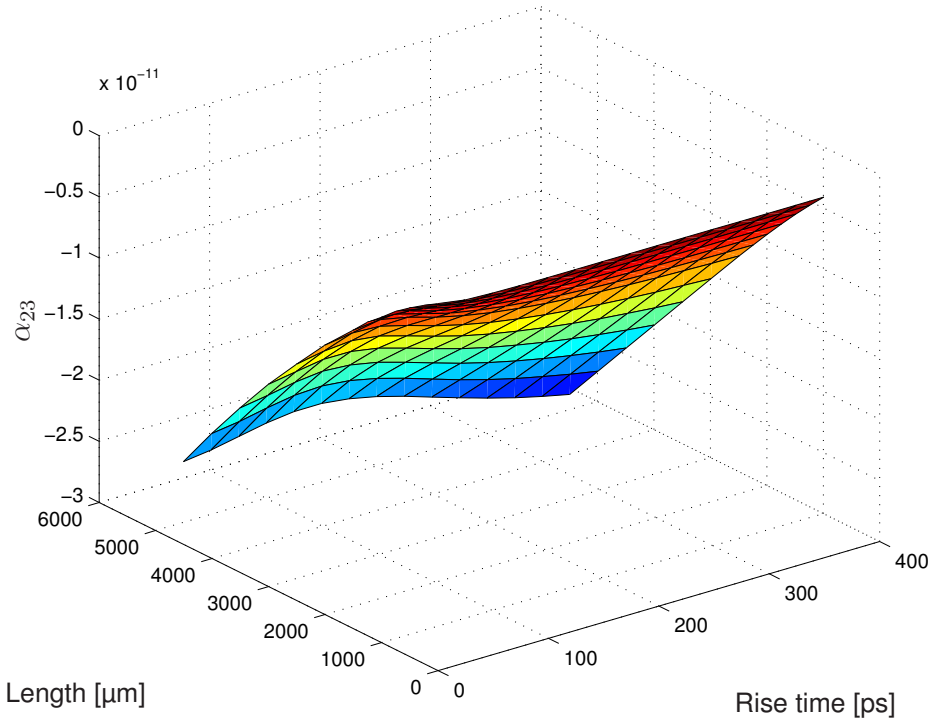


Fig. 4.10: α_{23} as a function of t_r and l in an RC-modeled 5-bit wide bus

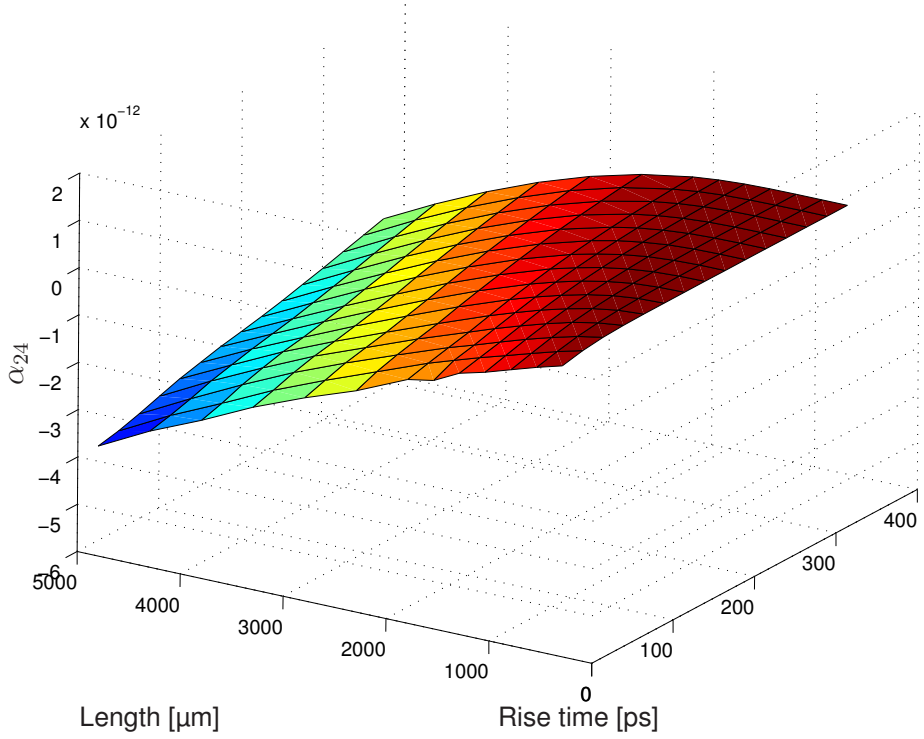


Fig. 4.11: α_{24} as a function of t_r and l in an RC-modeled 5-bit wide bus

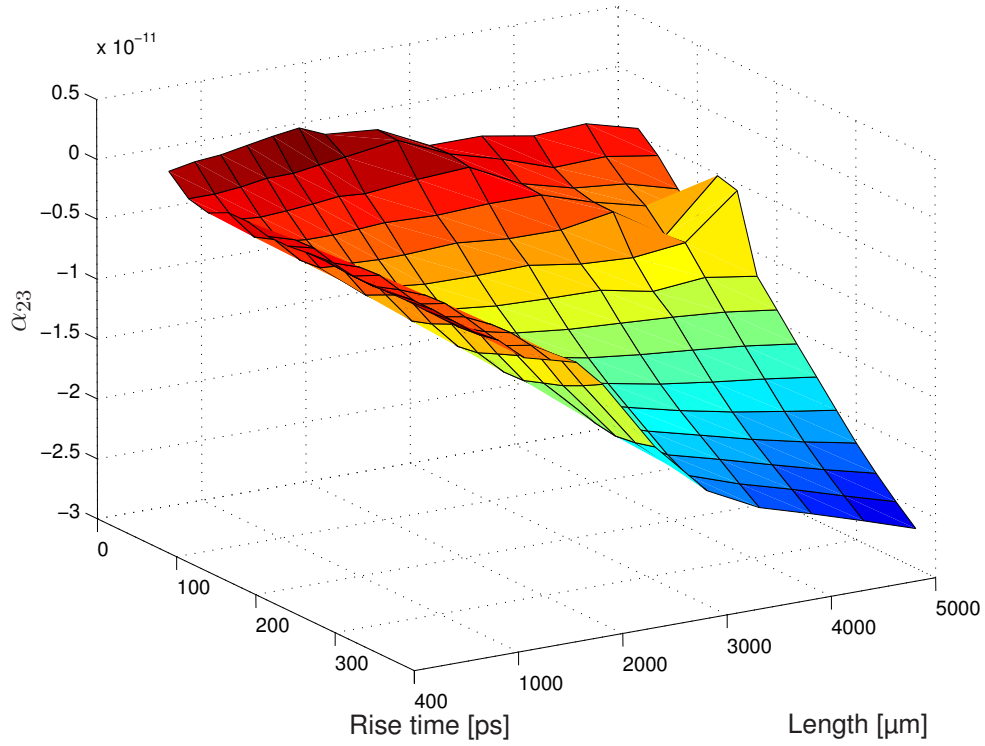


Fig. 4.12: α_{23} as a function of t_r and l in an $RLMC$-modeled 5-bit wide bus

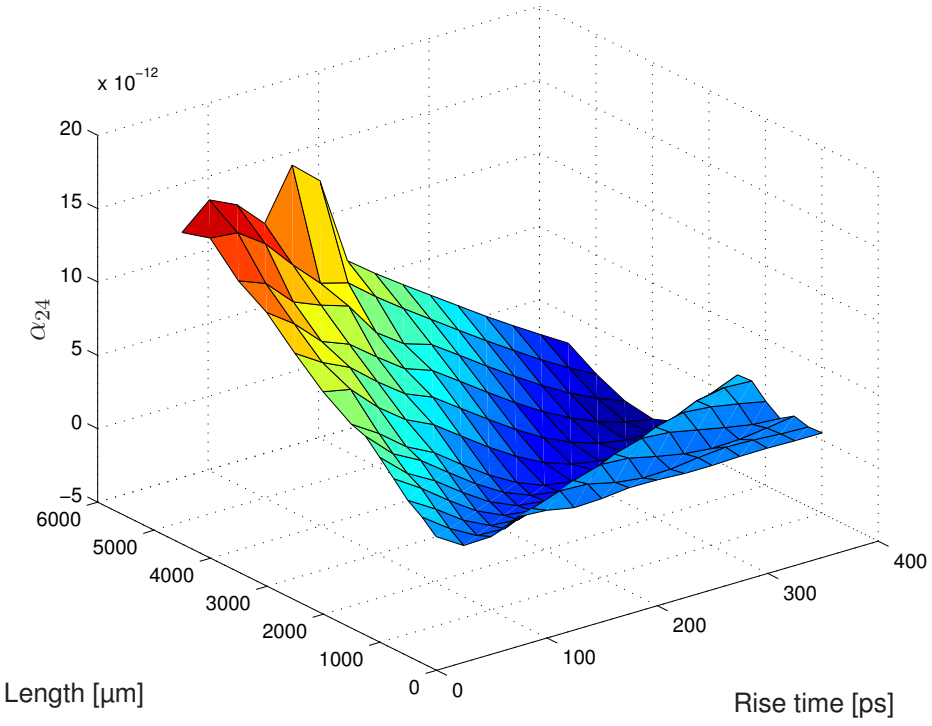


Fig. 4.13: α_{24} as a function of t_r and l in an $RLMC$-modeled 5-bit wide bus

4.2.3 Impact of Process Variations

As process technologies scale, the variations in several process parameters are continuously increasing and affecting more and more performance and power, as well as other parameters. For simulations where the effect of process parameter variations is included via Gaussian distributions, these values have been used as the expected value with a standard deviation specified as 0.1 µm. As explained later, we assume for the sake of simplicity that the variations are uncorrelated.

Variations have been classically divided in two categories: inter-die (die-to-die) and intra-die (within-die) variations. The inter-die variations are usually assumed to have a Gaussian distribution and when a number of process parameters are considered simultaneously it is important to take into account the correlation between these parameters. When device parameters vary within a single die as a function of their location, we talk about intra-die variations. Depending on the source of variations, within-die variations may be spatially correlated or uncorrelated and generally, modeling intra-die variations results in a huge complexity. Briefly, one can also say that variations are either spatially uncorrelated or correlated. Depending on their correlation distance, correlated variations are of inter-die or intra-die nature [183].

In order to model intra-die variations, a huge number of random variables is required. Some techniques have been developed in order to simplify analysis techniques when dealing concomitantly with correlated and independent sources of variations. For example, the Principal Component Analysis (PCA) is a statistical technique that maps a given set of correlated random variables to another set of uncorrelated random variables. The latter are called principal components, they are independent random variables, and the first few capture the most of the variability [183]. PCA-based techniques are used to simplify the correlation structure of variations in process parameters across a chip.

For simplicity, we choose to model thus variations as uncorrelated normally distributed random variables like proposed for inter-die variations in [183]:

$$\phi = \phi_{nom} + \Delta\phi, \tag{4.31}$$

where ϕ_{nom} is the nominal value of the process parameter and $\Delta\phi$ is a zero-mean random variable that captures uncorrelated variations. We model variations in the width and thickness of the interconnect via Gaussian distributions with a standard deviation of 0.1 µm. This variations have an important impact on all PUL parameters. It is to be noticed that the PUL parameters become thus random variables that generally do not follow a Gaussian distribution.

In order to prove the suitability of the aforementioned technique, we have constructed a model for the simplified scheme for process variations as previously presented. Nonetheless, the approach is not limited just to that scheme. We have generated 1000 uncorrelated sets of Gaussian distributed values for pitch, width, and thickness. For each of this 1000 sets, we have performed the extraction of the $RLMC$ parameters. Afterwards, we have

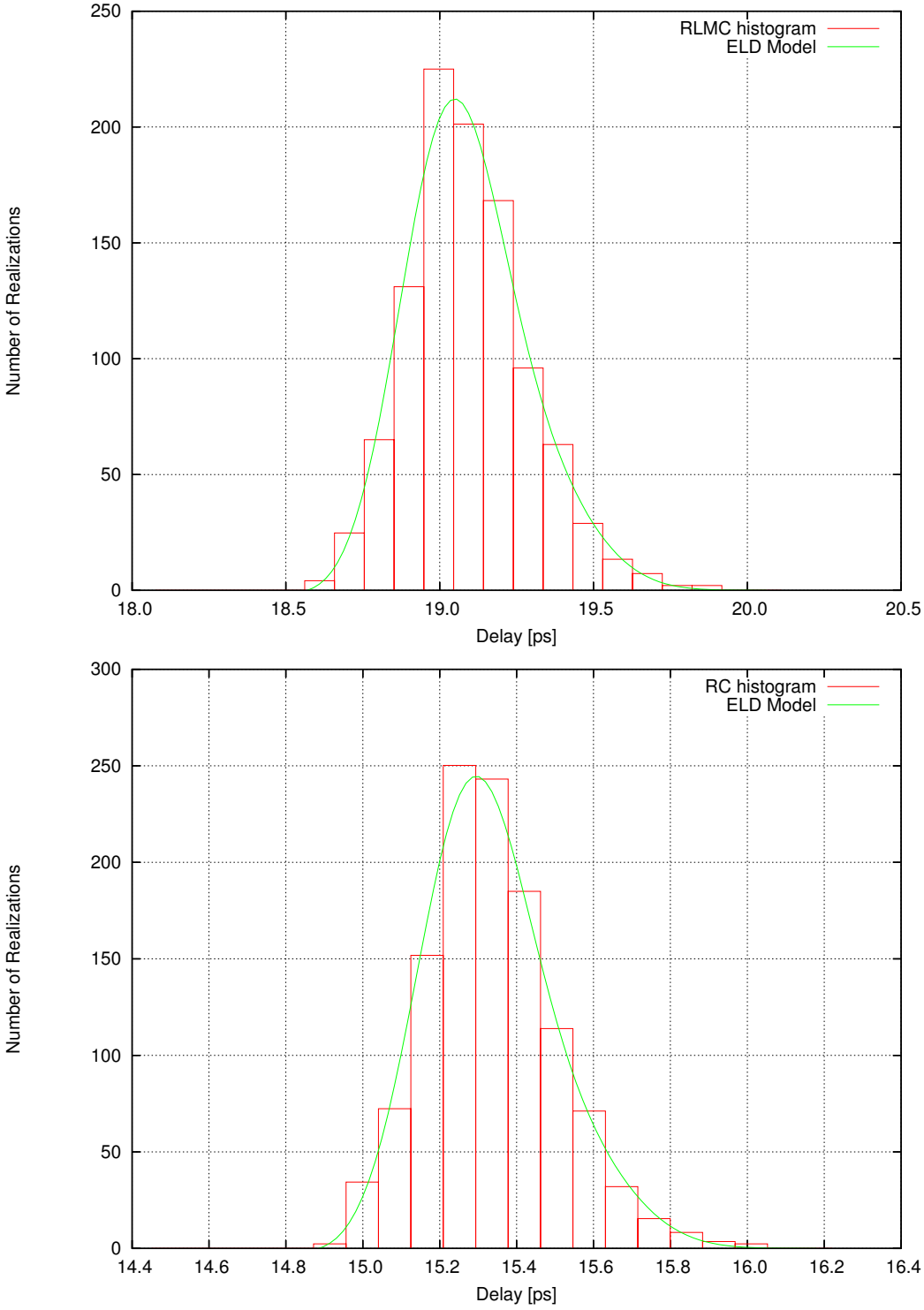


Fig. 4.14: Simulated and estimated delay under process variations

completed for each set SPICE simulations for calculating the delay tables and the corresponding coefficients, i.e. the A matrices.

The goal of this work is to provide the designer with a high-level model which can be used to abstract completely the physical world. By letting the model coefficients be random variables, it is possible to model in a compact way the effects of process variations

on delay. **Eq**. (4.30) can be thus rewritten in order to include process variations:

$$\underline{\delta} + \underline{\Delta\delta} = B \cdot (A + \Delta A) \cdot \underline{\Delta b}, \tag{4.32}$$

where $\underline{\Delta\delta}$ and ΔA represent the variation in delay and model coefficients respectively. In our scenario, the random variables are modeled as independent processes. Thus, only the one-dimensional probability function of each coefficient must be determined.

In **Fig**. 4.14, it can be noticed that the histogram of the coefficients is very close to a Gaussian. However, since all the PUL parameters but the resistance are skewed and non-linear function of the interconnect dimensions, the delay does not follow a Gaussian distribution. The probability density function (PDF) of a random variable can be accurately estimated by computing the first few moments [139]. Hence, this approach can be used for a fast yet efficient and accurate PDF estimation method. Thus, considering that only the first s moments are employed, each α_{ij} of the ELD model is replaced by a set of s moments $\{\mu_k^{(\alpha_{ij})}\}_{k=\overline{1,s}}$, where $\mu_k^{(\alpha_{ij})}$ is the *k*-th moment of the random variable α_{ij}. The moments of the δ_k-s and their PDF can be easily calculated from the ELD as shown in **Eq**. (4.32).

An expansion of the probability function in terms of Hermite polynomials [167] is very suited for close-to-Gaussian distributions as in the current scenario. By fitting just the first three moments of the random variable, we get the following approximation for the distribution $f_\alpha(x)$ of an α-coefficient:

$$f_\alpha(x) = \frac{(x^3 - 3x)\gamma_3 + 1}{\sqrt{2\pi}} \cdot \exp\left(-x^2/2\right), \tag{4.33}$$

where γ_3 is the skewness of the original α.

Consequently, in order to include the effects of process variations in our scenario, we have to characterize the model coefficients not only by their mean value, i.e. the first moment, but also by the second and third moments (standard deviation and skewness, respectively). In the case when the PDF differs significantly with respect to a Gaussian distribution, the approximation with Hermite polynomials becomes poor. In this case, high accuracy can be obtained by using more moments with expansions of the PDF in terms of other polynomials like Legendre or Laguerre [52].

4.3 Modeling of Power Consumption in Interconnects

Classically, the energy dissipated in on-chip interconnects has been determined only by the ground capacitance, also known as self or line capacitance. Thus, the energy dissipated by the bus drivers for charging the parasitic interconnect capacitances has been considered directly proportional to the number of bit transitions in the bus [128, 142]. Nevertheless, as previously mentioned, the coupling capacitance cannot be ignored for an accurate estimation and modeling of the dynamic power consumption component in buses because of continuously increasing aspect ratios in VDSM buses.

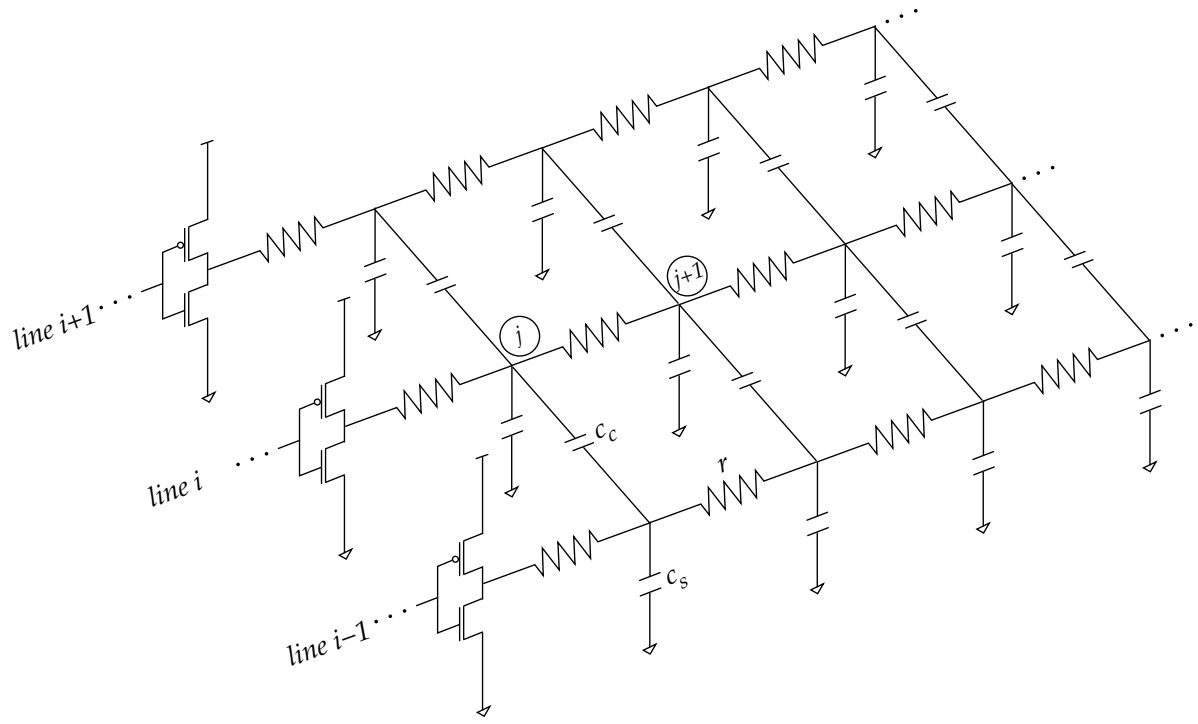


Fig. 4.15: RC model of a VDSM bus

In the sequel, a bus power macromodel is constructed that takes into account the inter-wire capacitance. As discussed in **Sec**. 2.3, neither self nor mutual inductances affect the overall dynamic power consumption. Therefore, inductances are neglected for the purpose of power macromodeling and the resulting interconnect model is depicted in **Fig**. 4.15, where j indicates the segment number in a line, c_s is the PUL self capacitance, and c_c represents the PUL coupling capacitance, while C_s and C_c represent the total self and total coupling capacitance, respectively.

4.3.1 Self, Coupling, and Equivalent Transition Activity

Let b_i^+ and b_i^- be the current and previous, respectively, digital value on the i-th line, and let $\Delta b_i = b_i^+ - b_i^-$ be the bit level self transition in line i. When $b_i^+ = 0$, only the NMOS transistor of the line driver is active and ideally no energy is drawn, and when $b_i^+ = 1$, all the current required for loading the capacitors is taken from the supply.

Let $v_{ij}(t)$ denote the voltage on the j-th segment in line i, and let S be the total number of segments required for an accurate interconnect model. Then, the total current through line i, $i_{ci}(t)$, can be calculated as:

$$i_{ci}(t) = \frac{C_s}{S}\sum_{j=1}^{N}\frac{\partial v_{i,j}}{\partial t} + \frac{C_c}{S}\sum_{j=1}^{N}\left(\frac{\partial v_{i,j}}{\partial t} - \frac{\partial v_{i-1,j}}{\partial t}\right) + \frac{C_c}{S}\sum_{j=1}^{N}\left(\frac{\partial v_{i,j}}{\partial t} - \frac{\partial v_{i+1,j}}{\partial t}\right) \tag{4.34}$$

This current is independent of the PUL resistance r but it is a function of the voltage variation in each section. As shown in [52], by assuming that the lines toggle synchronously

and with a complete swing of V_{dd}, **Eq**. (4.34) can be integrated to find the energy consumption over a period T. Thus,

$$\begin{aligned} E_i &= \int_0^T V_{dd} i_c(t) dt = \\ &= V_{dd}^2 \left[C_s \, \Delta b_i + C_t \, (2\Delta b_i - \Delta b_{i-1} - \Delta b_{i+1}) \right], \end{aligned}$$

while the mean value is obtained by applying the expectation operator $\mathbf{E}[\cdot]$ [52]:

$$\hat{E}_i = \mathbf{E}[E_i] = \frac{V_{dd}^2}{2} \left[C_s \, t_{s\,i} + 2 C_c \, t_{c\,i} \right] \tag{4.35}$$

where:

$$t_{s\,i} = 2\mathbf{E}[\, b_i^+ \Delta b_i \,] \tag{4.36}$$

$$t_{c\,i} = \mathbf{E}[\, b_i^+ \, (2\Delta b_i - \Delta b_{i+1} - \Delta b_{i-1}) \,] = t_{s\,i} - \mathbf{E}[b_i^+ (\Delta b_{i+1} + \Delta b_{i-1})] \tag{4.37}$$

are the so-called bit-level self and coupling transition activity[2], respectively. Note that in [52], $t_{s\,i}$ and $t_{c\,i}$ are called temporal and equivalent spatial transition activity, respectively. Let $t_{eq\,i}$ be the equivalent transition activity in line i:

$$t_{eq\,i} = t_{s\,i} + 2\kappa t_{c\,i}. \tag{4.38}$$

Then, the mean average power consumption can also be written as:

$$\hat{E}_i = \frac{C_s V_{dd}^2}{2} \cdot t_{eq\,i}. \tag{4.39}$$

The previous equations indicate that the energy consumption related to the ground capacitance is always positive, and proportional to the number of low-high bit transitions. Nonetheless, the energy associated with the coupling capacitances can be either positive or negative, with a maximum value of $+4$ and a minimum of -2. This means, that in some scenarios for some lines, some current may be drawn from the voltage source connected through the pull-up network, while in others, the current may be pulled back [206]. Moreover, it can be noticed that $t_{s\,i}$ is equal to $t_{c\,i}$ when b_i, b_{i-1}, and b_{i+1} are mutually independent. Furthermore, $t_{c\,i}$ becomes zero when the bits are completely correlated.

It is important to notice that a bus can be isolated (with margins) or non-isolated (without margins) depending on the existence of two ground wires or a V_{dd}-Ground pair to the rightmost and leftmost of the bus [175]. If margins are considered, then $b_0 = b_{n+1} = 0$ (or 1) which means that $\Delta b_0 = \Delta b_{n+1} = 0$. Otherwise, in a non-isolated bus the leftmost and rightmost bits are duplicated for the above formulas to hold: $b_0 = b_1$ and $b_{n+1} = b_n$.

[2]Throughout this work, we also refer to $t_{s\,i}$ and $t_{c\,i}$ as self activity and coupling activity, respectively

Tab. 4.2: Energy consumption in asynchronously toggling coupled lines (after [52])

Simultaneous				Slower victim					Faster victim					
$b_1^- b_1^+$	$b_2^- b_2^+$	c_t	c_e	$b_1^- b_1^+ b_1^{++}$	$b_2^- b_2^+ b_2^{++}$	c_t	c_e		$b_1^{--} b_1^- b_1^+$	$b_2^{--} b_2^- b_2^+$	c_t	c_e		
0 1	0 1	1	0	0 0 1	0 1 1	1	**1**		0 1 1	0 0 1	1	0		
0 1	1 0	1	2	0 0 1	1 0 0	1	**1**		0 1 1	1 1 0	1	2		
1 0	0 1	0	0	1 1 0	0 1 1	0	**-1**		1 0 0	0 0 1	0	0		
1 0	1 0	0	0	1 1 0	1 0 0	0	**1**		1 0 0	1 1 0	0	0		

The total self transition activity, T_s, and the total coupling activity, T_c, are the sum over all n bus lines of the corresponding bit-level activities:

$$T_s = \sum_{i=1}^{n} t_{s\,i}, \tag{4.40}$$

$$T_c = \sum_{i=1}^{n} t_{c\,i} = T_s - \sum_{i=1}^{n} \mathbf{E}[b_i^+(\Delta b_{i+1} + \Delta b_{i-1})]. \tag{4.41}$$

Again, the mutual independence of neighboring triplets makes T_c equal to T_s. The average total energy consumption in a symmetric bus, $\hat{E}_t$, is then:

$$\hat{E}_t = \sum_{i=1}^{n} \hat{E}_i = \frac{C_s\, V_{dd}^2}{2} \cdot (T_s + 2\kappa T_c) = \frac{C_s\, V_{dd}^2}{2} \cdot T_{eq}, \tag{4.42}$$

where $T_{eq} = T_s + 2\kappa T_c$ represents the total equivalent transition activity.

4.3.2 Effect of Dynamic Delay

The previously considered assumption of simultaneous switching does not necessarily hold. **Tab**. 4.2 shows the multiplicative factors associated to the self and the coupling capacitances when two bus lines toggle simultaneously as well as asynchronously. It can be observed that a faster victim does not change its energy profile. However, a slower victim produces a variation in the consumed energy. Consequently, in order to accurately model the energy consumption when non-simultaneous toggling is possible, the relative delay between the switchings on neighboring lines must also be modeled.

As discussed in **Chap**. 3, in order to reduce delay in long interconnects, a common technique is to insert buffers into the lines. In the case of coupled lines however, if the buffers are not clocked, crosstalk-induced signal variations yield non-simultaneous line transitions. Thus, the effect described above, i.e. faster or slower victims, appears in buffered on-chip buses. The explanation is given by the effective capacitance seen by a toggling line. With a low-high transition in the victim line and one aggressor toggling in the opposite direction, the effective capacitance driven by the line driver corresponding

to that aggressor is much higher and therefore, the victim switches more slowly. This makes the toggling in the aggressor occur faster than that in the victim and the so-called dynamic delay effect appears [52]. The exact behavior depends actually on the toggling of the aggressors of aggressors, i.e. the neighbors of the aggressor lines, the so-called *aggressors of aggressors*.

The results of simulating a buffered bus structure in metal 3 of a 0.35 µm technology are given in **Fig**. 4.16. The aggressors toggle always in the same direction as the victim, while the toggling pattern of the aggressors of aggressors cover several cases. It can be observed, that at the fourth buffer a significant time shift appears. Hence, the dynamic delay implicates a systematic decrease in coupling activity, and thus in power consumption. This effect can be modeled empirically as a subtracted correcting value for t_{ci} and the approximate expression is given by:

$$t_b = \mathbf{E}[\,(b_{i+1}^{-} b_i^{+} b_{i-1}^{-} - b_{i+1}^{+} b_i^{-} b_{i-1}^{+})\,\Delta b_{i+1}\,\Delta b_i\,\Delta b_{i-1}\,] \tag{4.43}$$

The basic idea of the aforementioned approximation is to rectify the coupling activity for those cases where all the lines toggle in opposite directions.

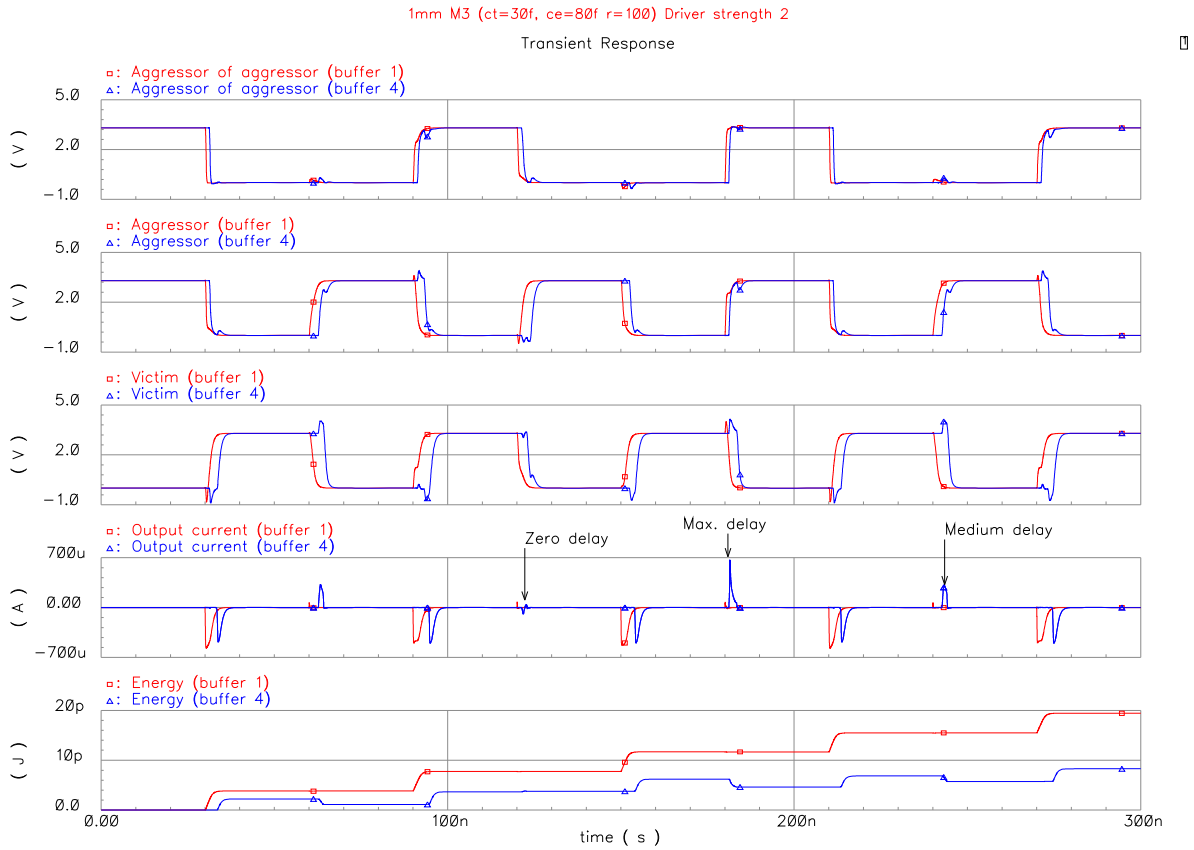


Fig. 4.16: Effect of dynamic delay in power consumption (after [52])

4.3.3 Inter-wire Coupling Activity

Relating the coupling transition activity – which is actually an inter-wire energy measure – to one single line, namely the driver that feeds that line, represents an elegant approach as the energy budget of each driver can be written in a simple fashion. Nevertheless, if a bus is not symmetric, the coupling capacitances of a victim line to the two neighbors are not necessarily equal anymore and the coupling has to be split and limited only to one neighbor. Thus, the coupling becomes a measure of the total energy consumption related to a pair of lines and not to a driver.

The ratio between the coupling capacitance and the self capacitance between two lines varies rapidly with small changes in the inter-wire spacing. Therefore, the bus aspect factor κ is defined only for a symmetric bus. The coupling capacitance varies much more rapidly than the self capacitance. In the following, the ground capacitance is considered for simplicity constant with spacing[3]. Thus, the self capacitance in every line is C_s, while the coupling capacitances between line i and line j, C_{cij}, vary as a function of spacing.

Let κ_{ij} be the bus aspect factor between line i and line j:

$$\kappa_{ij} = \frac{C_{cij}}{C_s}, \tag{4.44}$$

where κ_{ij} =0 if $|i-j| \neq 1$, and let $\kappa_i \stackrel{\text{def}}{=} \kappa_{ii+1}$ be the aspect factor between line i and line $i+1$. Let θ_{cij} be the inter-wire coupling activity between line i and line j:

$$\theta_{cij} = \mathbf{E}[(b_i^+ - b_j^+)(\Delta b_i - \Delta b_j)] \tag{4.45}$$

$$= \frac{t_{si}}{2} + \frac{t_{sj}}{2} - \mathbf{E}[b_i^+ \Delta b_j] - \mathbf{E}[b_j^+ \Delta b_i]. \tag{4.46}$$

and let $\theta_{ci} \stackrel{\text{def}}{=} \theta_{cii+1}$ be the inter-wire coupling between line i and line $i+1$. It can be easily verified that:

$$T_c = \sum_{i=1}^{n} t_{ci} = \sum_{i=0}^{n} \theta_{ci}, \tag{4.47}$$

where θ_0 and θ_1 are computed in function of the bus margins. Thus, the weighted total coupling activity, T_{Wc}, can be defined as:

$$T_{Wc} = \sum_{i=0}^{n} \frac{C_{cii+1}}{C_s} \cdot \theta_{ci} = \sum_{i=0}^{n} \kappa_i \, \theta_{ci}. \tag{4.48}$$

In order to correctly determine the total equivalent coupling activity, T_{Wc} must replace the factor κT_c in **Eq**. (4.42):

$$\hat{E}_t = \frac{C_s V_{dd}^2}{2} \cdot (T_s + 2T_{Wc}) \,. \tag{4.49}$$

Furthermore, even a theoretically symmetrical bus cannot be treated as such in the case of significant process variations.

[3]In reality, spacing influences the partitioning of the fringing capacitance between the total self and the coupling capacitances

4.4 Summary

In order to address timing issues at higher levels of abstraction, accurate models capable to predict pattern-dependent signal delay are required. This analysis is mandatory if delay-aware coding is to be employed. Previous techniques in that direction, such as [175, 181], are restricted only to non-inductively coupled interconnects because of a lack of proper models. Some efforts have been done to identify worst case switching patterns in inductively coupled lines [192]. However, those methods cannot predict the delay for a given input switching pattern. Further, the delay coding limits and delay elimination methods developed in [175] for capacitive coupling do not hold in the more general case of inductively-coupled lines.

In this chapter, an extended linear model for high-level signal delay estimation in both inductively and capacitively coupled on-chip buses has been constructed. The developed model approximates the signal delay as a linear combination of the contributions induced by each aggressor line for the complete set of switching patterns and not only for capacitively coupled buses or the worst case patterns. Root mean square errors less than 2 % have been reported. Therefore, the ELD model is suitable for fast yet efficient high-level analysis of bus encoding schemes focused on delay minimization in capacitively and inductively coupled lines.

The model has been extended to include the effects of process variations. For a simplified scheme, it has been proved that by considering the coefficients of the model as random variables and employing their first few moments, an accurate description of the delay variation can be obtained. The accuracy of the model has been assessed by means of extensive (more than 700.000) experiments employing state-of-the-art 3D capacitance and inductance extraction tools and SPICE simulations.

Furthermore, a power macromodel has been constructed that takes into account not only self capacitances but also inter-wire coupling capacitances. The essence of the macromodel is that in order to approximate and reduce the dynamic power consumption, the so-called self and coupling transition activities have to be estimated and decreased, respectively, at higher levels of abstraction. Moreover, it has been shown how to incorporate also the effects of dynamic delay. Finally, the notion of total coupling transition activity has been generalized to the so-called weighted total transition activity to encompass also the cases of non-symmetrical buses and process variations.

Chapter 5

Low-Power Coding in DSP Buses

Contents

As mentioned **Chap**. 3, in order to efficiently reduce power consumption in a bus by decreasing the associated transition activity, one has to employ simple codes, which in their turn should be all but power-hungry. Such low-power coding schemes have to be developed from an application-specific point of view rather than from a general perspective. In this work, the focus lies on DSP architectures due to their wide span of application domains. As shown below, both self and coupling transition activities in DSP

signals exhibit several properties of interest which can be exploited for developing simple yet efficient codes.

The main objective of this chapter is to construct simple yet effective low-power codes that reduce the self and coupling transition activities in both dedicated and shared interconnect structures of DSP architectures. The central idea is that by rigorously analyzing the bit-level and total transition activity, we can easily categorize the bits of a DSP bus in two regions with different statistical properties. Simply put, the most significant bits are strongly correlated while the least significant ones tend to be uncorrelated and uniformly distributed. Thus, at least two different coding sub-schemes have to be applied for appropriately tackling the problem of transition activity reduction. One the one hand, simple non-redundant codes can exploit and preserve the high correlation in the MSBs and on the other hand, bus-invert schemes are best-suited for uncorrelated uniformly distributed data. The proposed codes for dedicated buses can be easily extended to shared architectures by adding only one redundancy line.

This chapter is organized as follows: **Sec**. 5.1 analyzes the transition activity in DSP buses and gives a brief overview of existing models for self and coupling activity. In **Sec**. 5.2, the effectiveness of Bus Invert and Odd/Even Bus Invert is analyzed. **Sec**. 5.3 illustrates the way non-redundant codes manage to exploit the high correlation typical for DSP signals. Afterwards, low-complexity codes based on partial bus invert and odd/even bus invert are constructed and analyzed in **Sec**. 5.4. Those encoding schemes are extended to shared buses and data with time-varying characteristics. Eventually, **Sec**. 5.5 gives some theoretical limits regarding coding for power reduction.

5.1 Transition Activity in DSP Signals

The analysis, modeling, and estimation of the bit-level transition activity based on the word-level statistics and characteristics has been widely studied in the literature. In [94, 93], Landman and Rabaey proposed a piecewise-linear model for the self transition activity. The main disadvantage is however, the requirement of RTL (Register Transfer Level) simulations for extracting a set of parameters. Later, Ramprasad, Shanbhag, and Hajj surmounted that drawback by extending the framework to an analytical one [145, 146]. Nevertheless, the technique provides only a rough approximation of the transition activity in the most significant bits. Therefore, Bobba, Hajj, and Shanbhag developed in [17] a more accurate model which is nonetheless restricted to zero-mean Gaussian signals. By describing the transition activity problem from a mathematically rigorous point of view, García et al. managed to overcome the limitations of the previously mentioned models and provided an accurate model for both self and coupling transition activity and non-zero mean signals [52].

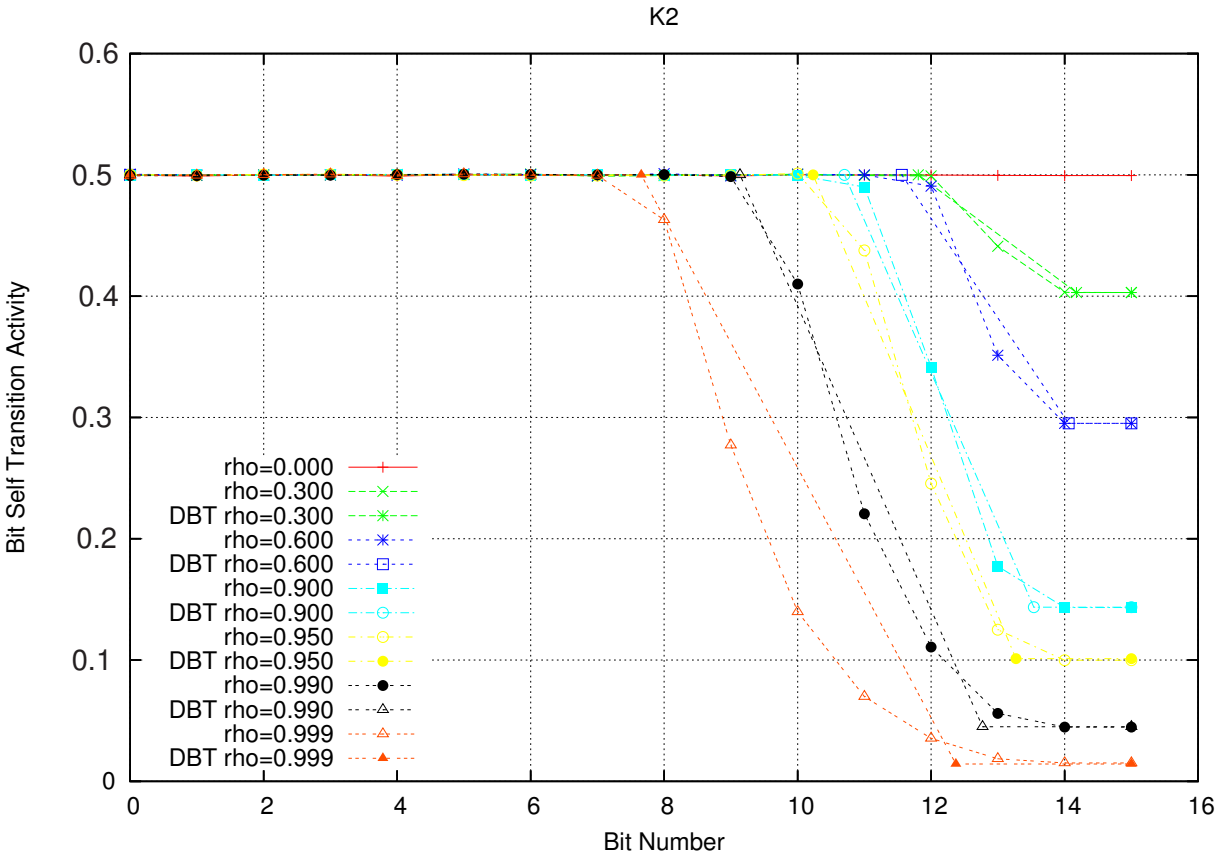


Fig. 5.1: Bit-level self activity: K2, $B = 16$, varying ρ, $\sigma_n = 0.12$

The efficient estimation and modeling of the transition activity is outside the scope of this work. However, in order to develop simple yet efficient low-power codes for DSP signals a thorough understanding of the bit-level transition activities is required. Therefore, we focus in the following on those models which underline the most important characteristics required for designing simple yet efficient codes.

5.1.1 The Dual-Bit Type Model

After analyzing the shape of the bit-level self transition activity in common DSP signals represented in two's complement (K2), Landman and Rabaey showed that the transition activity can be approximated by means of a continuous piecewise linear model [93, 94]. Basically, the model consists of three regions defined by two so-called breakpoints, namely BP_0 (or the LSB breakpoint) and BP_1 (or the MSB breakpoint). The regions correspond to the behavior of the most significant bits, the intermediate bits, and the least significant bits, respectively. The robustness of the abovementioned formulas for uni-modal distributions has been observed, even though they were derived mainly for Gaussian signals [94].

In the following, σ and ρ denote the standard deviation and the correlation of the analyzed signal, respectively. It is to be noticed, that instead of σ, one can use the so-called normalized standard deviation defined as:

$$\sigma_n = \frac{\sigma}{2^{B-1}}, \tag{5.1}$$

where B is the bus width. The normalized standard deviation is a measure of the relative value of σ with respect to the total signal range and is used in order to envelop signals of different width under a common frame [52].

Fig. 5.1 shows the simulatively obtained and modeled self transition activity in synthetic DSP signals represented in K2. In order to generate the data, we have employed the Gaussian ARMA (Auto-Regressive Moving Average) model driven by a white, zero-mean Gaussian noise introduced in [145, 146]. That Gaussian ARMA model can be regarded as a linear IIR filter that colors the white Gaussian noise at the input in order to achieve a certain word-level temporal correlation.

The bits in the LSB region tend to be uncorrelated and uniformly distributed and can be modeled as such. On the contrary, the bits in the MSB region can be strongly correlated. Their simultaneous toggling can be captured by another parameter which depends on the signal characteristics called, namely the self transition activity in the MSB, t_m. The bit-level self activity can be modeled thus as 0.5 in the LSBs and as t_m in the MSBs. The linear DBT (Dual-Bit Type) model can be formulated as:

$$t_i = \begin{cases} \frac{1}{2} & \text{when} \quad i \leq BP_0 \\ \frac{1}{2} + \frac{(i - BP_0)\,(t_m - \frac{1}{2})}{BP_1 - BP_0} & \text{when} \quad BP_0 < i < BP_1 \\ t_m & \text{when} \quad i \geq BP_1 \end{cases} \tag{5.2}$$

where t_i represents the activity in the i-th bit. The two breakpoints can be empirically approximated by:

$$BP_0 = \log_2 \sigma + \Delta BP_0, \tag{5.3}$$

$$\Delta BP_0 = \log_2 \left(\sqrt{1 - \rho^2} + \frac{|\rho|}{8} \right), \tag{5.4}$$

$$BP_1 = \log_2 (|\mu| + 3\,\sigma), \tag{5.5}$$

where ΔBP_0 represents a correction factor that is added to BP_0. For an explanation of the formulae for the breakpoints, the interested reader is referred to [93, 94].

Landman and Ramprasad developed the DBT model only for self activity, but as shown in the sequel, the model can be easily extended for coupling activity. **Fig**. 5.2 shows the simulated and modeled bit-level coupling transition activity for the same set of synthetic signals as above. We can notice that the shape is similar to the one of the self activity. However, there are two mentionable differences. First and most significantly, the coupling activity in the MSBs is zero even for small values of the correlation factor and

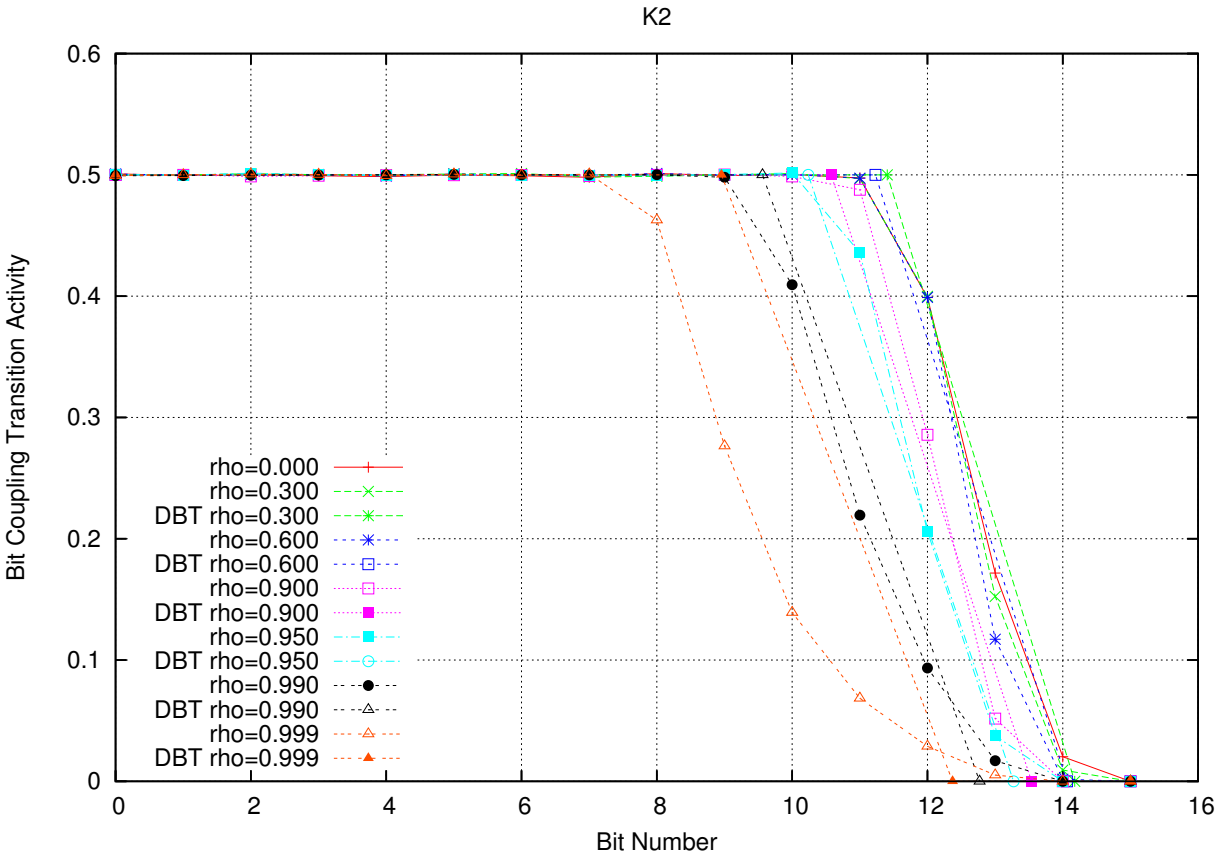


Fig. 5.2: Bit-level coupling activity: K2, $B = 16$, varying ρ, $\sigma_n = 0.12$

secondly, the breakpoints are slightly different. Thus, an extended version of the DBT model can be constructed by including correction factors for both breakpoints:

$$\Delta BP_k = log_2\left(\frac{\sqrt{1-\rho^2}}{\alpha_k} + \frac{|\rho|}{\beta_k}\right), \tag{5.6}$$

where k=1, 2. The coefficients α_k and β_k are functions of ρ and σ_n, which can be derived empirically in a similar way as indicated in [94]. Moreover, the coupling activity in the MSBs is zero and must not be simulatively or analytically determined.

5.1.2 Analytical Model for the Transition Activity

Ramprasad et al. reformulated in [145,146] the problem of modeling the transition activity in terms of bit probability, $p_i = \mathbf{E}[b_i]$, and bit-level temporal correlation, ρ_i, by employing a theoretically exact expression:

$$t_i = p_i\,(1 - p_i)\,(1 - \rho_i). \tag{5.7}$$

Similar to Landman and Rabaey, the authors proposed a piecewise linear model for modeling the bit-level correlation. The bit-level correlation can be analytically approxi-

mated by means of three parameters, namely the breakpoints (BP_0, BP_1) and the correlation in the MSB (ρ_m):

$$\rho_i = \begin{cases} 0 & \text{when} \quad i < BP_0 \\ \dfrac{i - BP_0 + 1}{BP_1 - BP_0}\,\rho_m & \text{when} \quad BP_0 \leq i < BP_1 - 1 \\ \rho_m & \text{when} \quad i \geq BP_i - 1 \end{cases} \tag{5.8}$$

The bit-level correlation in the MSB has been approximated by the word-level correlation, i.e. $\rho_m = \rho$. This can induce significant errors especially for buses with a wide MSB region.

However, Bobba, Hajj and Shanbhag derived in [17] a theoretical exact expression for the transition activity in the MSB in zero-mean Gaussian signals:

$$t_m = \frac{\text{acos}(\rho)}{\pi}. \tag{5.9}$$

When a signal exhibits a non-zero mean value, t_m cannot be analytically calculated with the abovementioned formula. In order to overcome this limitation, an analytical framework has been developed in [52]. Thus, formulas have been developed for correlated, anticorrelated and poorly correlated signals.

In **Fig**. 5.1 and **Fig**. 5.2 it can be observed that for high correlation factors, the piecewise-linear model looses its accuracy. Therefore, as proposed in [52], it is more attractive to model directly the total self transition activity and total coupling transition activity. As a consequence, the following formulas have been obtained:

$$T_s \approx \frac{B}{2} + dt_m\ \log_2(3.4\,\sigma_n) - 0.73\,dt_m\sqrt{\frac{dt_m}{t_m}} \tag{5.10}$$

$$T_c \approx 0.5\log_2(3.25\,\sigma) - 0.48\,\frac{dt_m}{\sqrt{t_m}} \tag{5.11}$$

The parameter dt_m represents the difference between the transition activities of the least and the most significant bits and is referred to as (transition) activity excess [52]:

$$dt_m \stackrel{\text{def}}{=} 0.5 - t_m = \frac{\text{asin}(\rho)}{\pi}. \tag{5.12}$$

The abovementioned analytical models allow to estimate the shape of the self and coupling transition activities at high levels of abstraction in DSP signals. Thus, when disposing of this information *a priori* to the explicit system design and implementation, one can easily choose for each bit region the most appropriate encoding scheme. Furthermore, when the data characteristics are unknown at design time, the regions have to be accurately estimated for an adequate coding solution.

5.2 Analysis of Bus Invert Coding Schemes

As previously showed, the bit-level coupling activity in the MSBs is zero and thus, in order to reduce the total coupling activity only the bit-level coupling activity in the LSBs must be minimized. However, for an efficient optimization of the total self activity, the bit-level activity in the MSB region also has to be scaled down depending on the value of t_m. It is nonetheless important to mention here, that a low-power code which is effective for reducing the total self activity in the MSBs may considerably increase the corresponding total coupling activity. Thus, we can state that the coding for the MSB and LSB regions has to be performed generally in a disjunctive manner.

Furthermore, it is of utmost importance to develop encoding schemes with as few as possible redundancy bits. Otherwise, the power consumption associated to the coupling activity may increase dramatically as a higher number of bus lines must be spaced more closely. In addition, due to the fact that bus invert is the most efficient 1-bit redundant code for uncorrelated uniformly distributed data [187,189], bus invert based schemes are very attractive to be employed for reducing the transition activity in the LSBs.

In the following, we analyze the effects of the bus invert (BI) scheme [187] and the 2-bit redundant odd/even bus invert (OEBI) code developed in [206]. As the main central idea is to reduce the activity in the LSB region, we focus on uncorrelated uniformly distributed data.

5.2.1 Self Transition Activity

Lin and Tsai performed in [99] a theoretical analysis of the Hamming-distance based bus invert applied on uncorrelated uniformly distributed data. The obtained formulas have been afterwards used to analyze the impact of partitioning a bus into smaller sub-buses and to show the viability of BIH as a function of the bus width. The mean self transition activity per bit can be expressed thus as a function of the bus width n:

$$T_s(n) = \begin{cases} \dfrac{1}{2} - \dfrac{1}{2^{n+1}} \cdot \binom{n}{\frac{n}{2}}, & \text{for even bus width} \\ \dfrac{1}{2} - \dfrac{1}{2^n} \cdot \dfrac{n}{n+1} \cdot \binom{n-1}{\frac{n-1}{2}}, & \text{for odd bus width} \end{cases} \tag{5.13}$$

where $n+1$ is the bus width including the bus invert, and $\binom{n}{k}$ represents the number of combinations of n things taken k at a time. Due to the fact that:

$$n\binom{n-1}{k-1} = k\binom{n}{k}, \tag{5.14}$$

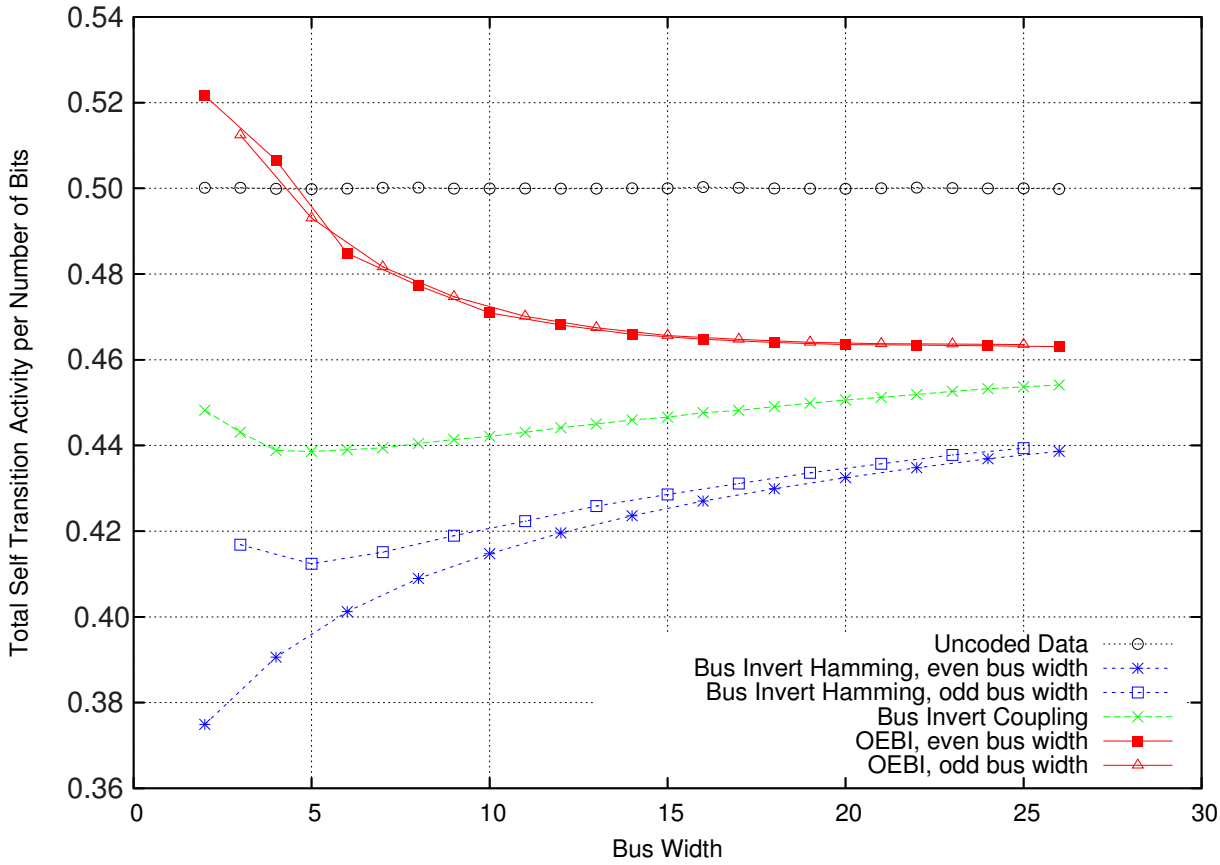


Fig. 5.3: Effects of Bus Invert schemes on total self activity per bit

Eq. (5.13) can be simplified as follows:

$$T_s(n) = \frac{1}{2} - \frac{1}{2^{n+1}} \cdot \binom{n}{\lfloor \frac{n+1}{2} \rfloor}, \tag{5.15}$$

where $\lfloor \cdot \rfloor$ represents the floor function. Similar equations have been derived for coupling reduction analysis in [98].

Fig. 5.3 illustrates the mean self activity for uncoded data and for data coded with BIH, BIC and OEBI. The activity has been determined simulatively on one million uncorrelated and uniformly distributed samples for each bus width. First, we can observe that for BIH, the mean self activity differs for even and odd bus widths and that the effectiveness of BIH decreases with increasing bus width, as predicted by the aforementioned formulas.

Because the problem of reducing the self activity and that of decreasing coupling activity have a certain degree of correlation, we can notice that BIC also manages to reduce the self activity, nonetheless at a lower rate. Due mainly to its 2-bit redundancy, OEBI performs poorly especially for low values of the bus width.

5.2.2 Coupling Transition Activity

In general, inverting an entire bus in order to reduce the Hamming distance also implies a decrease in the coupling activity. As seen in **Fig**. 5.4, BIH has a similar effect on the mean coupling activity as on the mean self activity. Additionally, BIC reduces T_c as expected by a higher amount than BIH.

The most effective scheme for optimizing the coupling activity is OEBI which clearly outperforms BIC and BIH for any bus width higher than three. Furthermore, it also important to notice that the mean coupling activity even when applying OEBI for wide buses is lower than the resulting coupling activity when applied on narrow buses. Thus, in order to efficiently reduce the total coupling activity in wide buses, it is more appropriate to apply OEBI for the entire bus than applying BIC or BIH on a partitioned (clustered) bus. The extra degree of freedom introduced by the additional redundant bit increases the versatility of OEBI in comparison with BIC or clustered BIC. The main advantage of OEBI is that it treats the bus lines in an interleaved manner. Additionally, as noticed in the previous subsection for wider buses, OEBI reduces the gap with respect to BIC and BIH in terms of self activity.

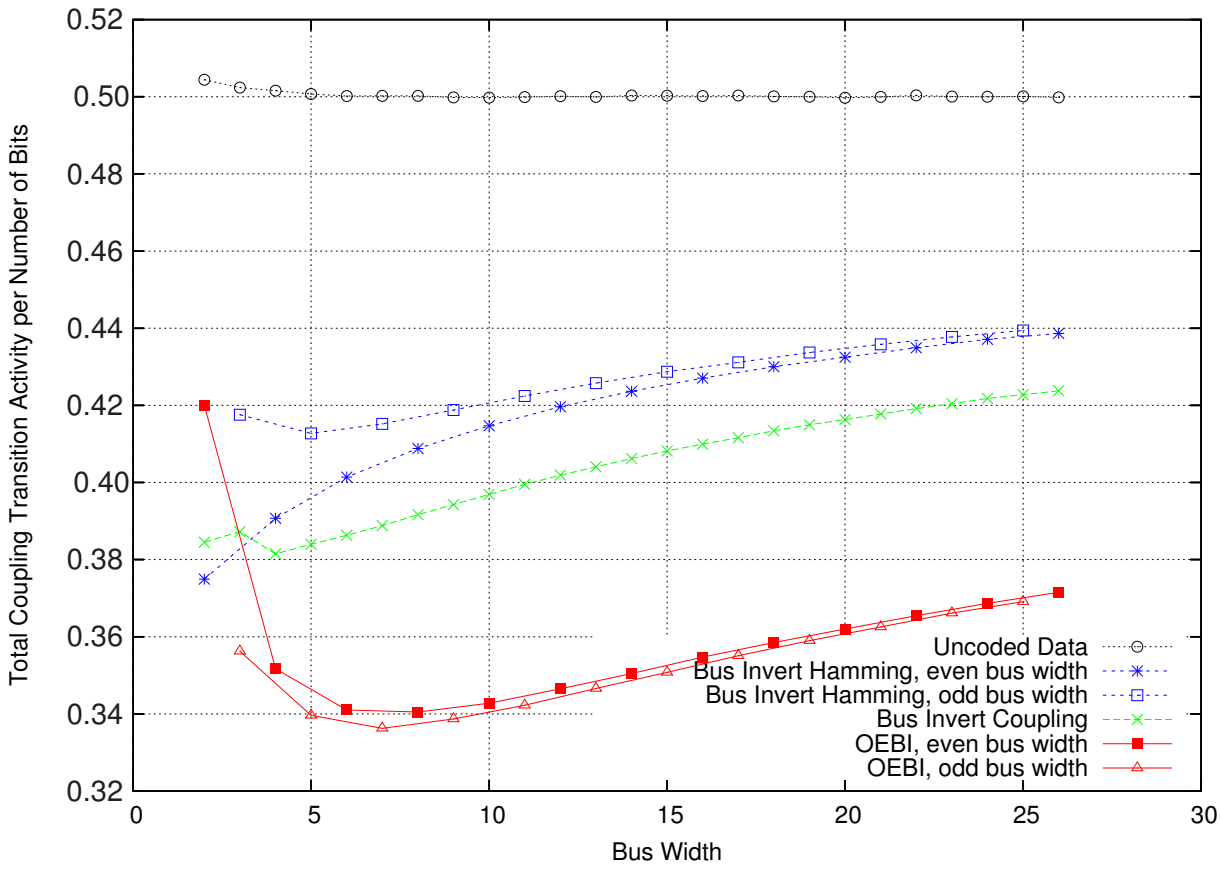


Fig. 5.4: Effects of Bus Invert schemes on total coupling activity per bit

It can be thus concluded, that in tightly coupled buses in which the coupling capacitances outweigh the self capacitance, it is of higher interest to employ OEBI or BIC instead of BIH. Generally, OEBI outperforms BIC, especially for higher values of the bus width.

5.3 Exploiting Temporal and Spatial Bit Correlation in DSP Buses

In **Sec**. 5.1, we have seen that in the most significant bits of common DSP signals, the bit-level self transition activity is not necessarily zero like in the case of the coupling transition activity. Therefore, in this section, we introduce and study several non-redundant codes aiming for reducing the self activity in the MSBs and the coupling activity in the intermediate bits. Moreover, we show how the classic bus invert schemes can be improved by combining them with non-redundant codes.

5.3.1 Transition Activity in Non-redundant Codes

As in **Sec**. 5.1, it is assumed that data is represented in 2's complement (K2). In addition, the following non-redundant codes are considered:

K1 In this code, each bit is XOR-ed with the MSB, i.e. $b_i^{'} = b_i \oplus b_n$. This code reduces the self transition activity in the higher bits for small σ-s. This encoding is similar to a single-zero sign-magnitude representation. The decoder is identical to the encoder.

K0 In this case, the transmitted bit is computed as the XOR of two neighboring bits, i.e. $b_i^{'} = b_i \oplus b_{i+1}$. The MSB is left unchanged. This code profits better from the spatial correlation between neighboring bits, though the decoding is slower as it requires a cascade of XOR gates. It is to be noticed that this code is equivalent to a Gray mapping.

K3 This encoding scheme is actually exactly the decoder of the K0 code, i.e $b_i^{'} = b_i \oplus b_{i+1}^{'}$. It exploits the spatial correlation between the neighboring bits when one of these is already encoded.

KP This scheme is actually a permutation and has been used only in conjunction with the other mentioned codes. This permutation has been proposed in [101] for address buses in conjunction with NK3 (the so-called G-Code).

Additionally, we also consider the following two non-redundant encoding schemes:

NK0 In this case, the transmitted bit is computed as the NXOR of two neighboring bits, i.e. $\overline{b_i^{'}} = b_i \oplus b_{i+1}$.

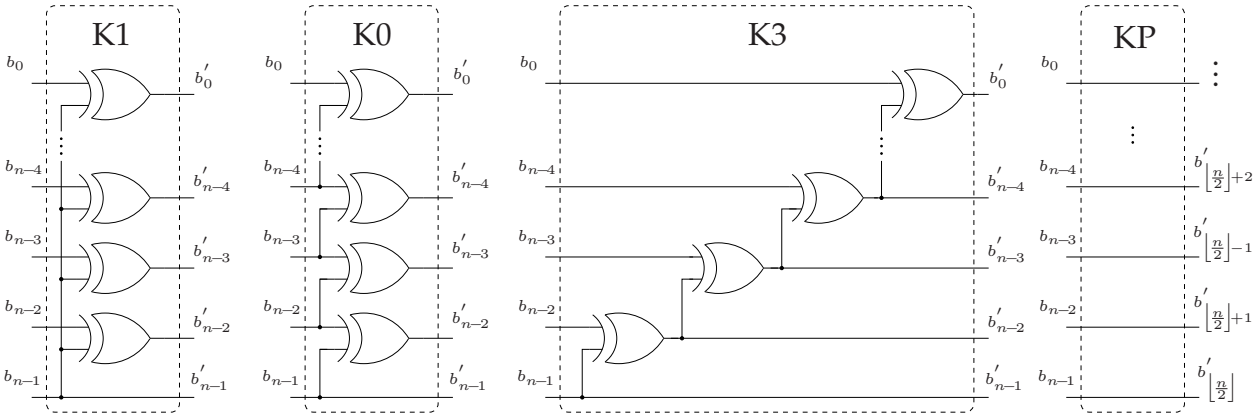


Fig. 5.5: Non redundant codes K0, K1, K3, and the permutation KP. $\lfloor \cdot \rfloor$ is the *floor* operator, b_i and $b_i^{'}$ represent the input and output bits respectively.

NK3 This encoding scheme is similar to K0 code, however with NXORs replacing XORs, i.e $\overline{b_i^{'}} = b_i \oplus b_{i+1}^{'}$.

All the aforementioned schemes are based on the fact that in the case of DSP signals, there is a high spatial correlation between neighboring bits. As shown in the sequel, the same schemes are able to exploit a high temporal correlation of each bit value, especially when combined with redundant codes.

Fig. 5.6 and **Fig**. 5.7 illustrate the effectiveness of the K0 code. The bit-level self activity in the most significant bits is reduced to zero by making use of the temporal correlation and the small existing spatial and temporal correlation in the intermediate bits is also exploited to slightly decrease the corresponding bit-level coupling activity. K3 and NK3 have similar effects as K0 and NK0, while the more simple K1 representation is less effective.

As mentioned in **Sec**. 5.3, we can write in the case of a non-redundant code a more general formula for the total coupling activities:

$$T_c \approx 0.5 \log_2(\alpha_c \sigma) - \beta_c \frac{dt_m^{\gamma_c}}{\sqrt{t_m}}, \tag{5.16}$$

where γ_c is restricted to a multiple of 0.5 for allowing an efficient determination of the equation [52]. From the abovementioned formula, we can see that the data characterized by a higher ρ and a smaller σ exhibit a lower transition activity than poorly correlated and widely spread data. Also, for a low-power data transmission, one has to employ codes for which α_c has a small value and β_c a high one. Moreover, we can write in the case of K1, K2, and K3, a similar equation for the total self transition activity, i.e.:

$$T_s \approx 0.5 \log_2(\alpha_s \sigma) - \beta_s \frac{dt_m^{\gamma_s}}{\sqrt{t_m}}, \tag{5.17}$$

where γ_c is again restricted to a multiple of 0.5 for the same computational reasons.

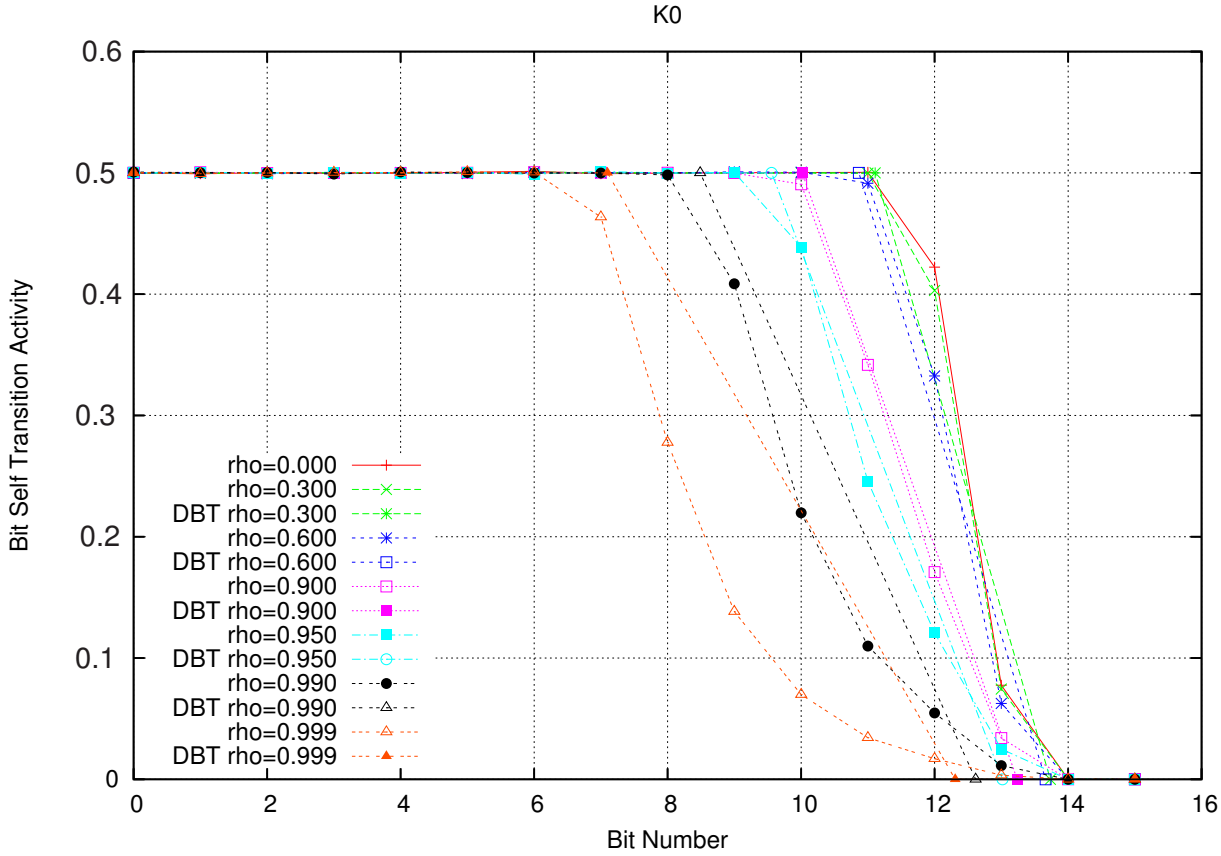


Fig. 5.6: Bit-level self activity: K0, $B = 16$, varying ρ, $\sigma_n = 0.12$

For instance, by fitting the experimental results, the following estimations have been obtained for K0:

$$T_s \approx 0.5 \log_2(2.10\,\sigma) - 0.80\,dt_m \sqrt{\frac{dt_m}{t_m}}, \tag{5.18}$$

$$T_c \approx 0.5 \log_2(1.72\,\sigma) - 1.05\,\frac{dt_m^2}{\sqrt{t_m}}, \tag{5.19}$$

and for K1:

$$T_s \approx 0.5 \log_2(1.90\,\sigma) - 0.84\,\frac{dt_m^2}{\sqrt{t_m}}, \tag{5.20}$$

$$T_c \approx 0.5 \log_2(2.15\,\sigma) - 1.15\,dt_m^2 \sqrt{\frac{dt_m}{t_m}}. \tag{5.21}$$

Basically, the two terms of the equation put in balance the effects of σ and ρ. As expected, the transition activity augments with increasing σ and decreasing ρ. We can also observe that K0 is more suitable for reducing the coupling activity than K1. However, for a set of certain (σ, ρ)-values, K1 is able to achieve a higher reduction in terms of self activity than K0.

Fig. 5.8 and **Fig**. 5.9 illustrate the coupling and the self activity for non-redundant codes. With regard to the coupling activity, it can be seen that K2 is insignificantly better

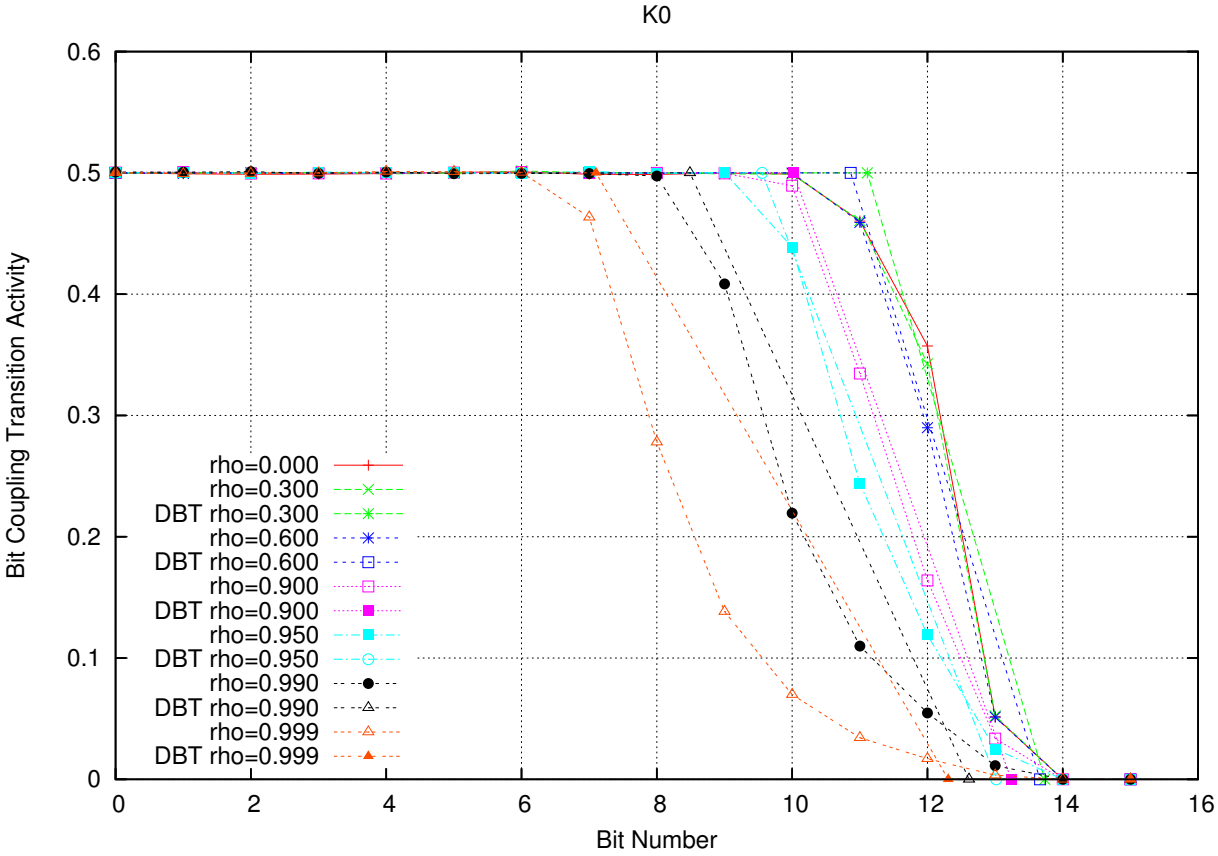


Fig. 5.7: Bit-level coupling activity: K0, $B = 16$, varying ρ, $\sigma_n = 0.12$

than K0 for very small correlations, in this case for $\rho < 0.2$. For ρ greater than 0.2, K0 gets more effective with increasing ρ. K3 achieves a smaller coupling activity than K2 only for very high correlated signals, and K1 does not manage to outperform K2 at any rate. Nevertheless, in the case of self activity, K1 is far better than K2. Further, K1 is comparable to K0 and significantly better than K3. For coupling activity reduction though, K1 does not represent an alternative to be taken into account.

5.3.2 Combining Non-redundant Codes and Bus Invert Schemes

The abovementioned non-redundant codes can be combined with redundant schemes like the bus invert based ones to obtain 1-bit or 2-bit redundant low-power codes. In the sequel, we refer for example to the cascading of K0 and OEBI as K0-OEBI.

Furthermore, we propose a 3-bit redundant code based on OEBI that outperforms OEBI despite the extra redundancy bit. The scheme implements K0-OEBI or K3-OEBI based on the coupling metric. There is an important advantage related to the codec architecture when integrating K0 and K3 in the same scheme. The rationale behind that is related to the fact that when implementing K0 or K3 on a bidirectional bus actually both scheme have to be included in the codec as they represent each other's decoder. There-

fore, both K0 and K3 can be employed together with virtually no hardware overhead. We call this code K0-OEBI&K3-OEBI or simply K0K3OEBI. Similarly, OEBI&K0-OEBI and OEBI&K3-OEBI have been constructed. Clustered versions of K0-OEBI (CK0OEBI) and K3-OEBI (CK3OEBI) have also been taken into consideration.

In order to perform a thorough analysis of the chosen codes, we first analyze how the codes perform on a huge set of synthetic data based on their statistical properties. For this purpose, we have applied the aforementioned codes on a huge number of zero-mean Gaussian distributed data with varying standard deviation and correlation factor. Afterwards, with the accumulated knowledge we apply various selected codes on a set of real DSP data.

At first sight, the XOR-based codes and the NXOR-based ones would seem to be having the same impact on transition activity. Though this is true for self transition activity, in the case of coupling transition activity, there is a difference related to the extreme bits which can slightly influence some redundant coding schemes. As mentioned before, the codes and the permutation try to exploit the temporal and spatial correlation and shift coupling transition that appear between adjacent lines to the extremes of the bus. The decision where to add the redundancy bits is therefore important as it can significantly influence – at least in a large amount of cases – the coupling transition activity in the MSB and/or LSB.

In order to analyze to which extent the codes can be optimized, we have chosen to perform simulations with all XOR- and NXOR-based codes for redundant schemes. More significant differences appear when there are at least two bits of redundancy. Therefore, we concentrate our attention in this matter on the modified OEBI schemes with two redundancy bits.

We have chosen to analyze signals represented in K2 with 8 and 16 bits. The normalized standard deviation has been varied between 0.02 and 0.2, while for the correlation factor a representative set between 0 and 0.999 has been selected. In the latter case, values between 0.90 and 0.99 are of high interest as they represent typical correlations in important DSP applications like audio data. It is to be noticed that as shown in [94], the case when the correlation is negative can be reduced to the positive one.

In **Fig**. 5.10 and **Fig**. 5.11 some results concerning 1-bit redundant codes are represented. In that case, BIC is in general the best encoding scheme. K0-BIC is more efficient only for a high correlation factor, i.e. $\rho > 0.85$ and K3-BIC shows its efficiency for $\rho > 0.96$. Things look different when analyzing the self activity. BIC is even worse than K2 for very high correlations and K0-BIC, K1-BIH and BIH outperform K2 by approximately 25%. A very interesting result is that K0-BIC is at any instance as efficient as BIH, but for high correlation signals where it is around 5% better.

Fig. 5.12, **Fig**. 5.13, **Fig**. 5.14, and **Fig**. 5.15 deal with the modified OEBI codes, i.e. schemes with two bits of redundancy. Again, the K0-based OEBI schemes outperform all the others. Another interesting result is that the classical OEBI is in terms of self activity

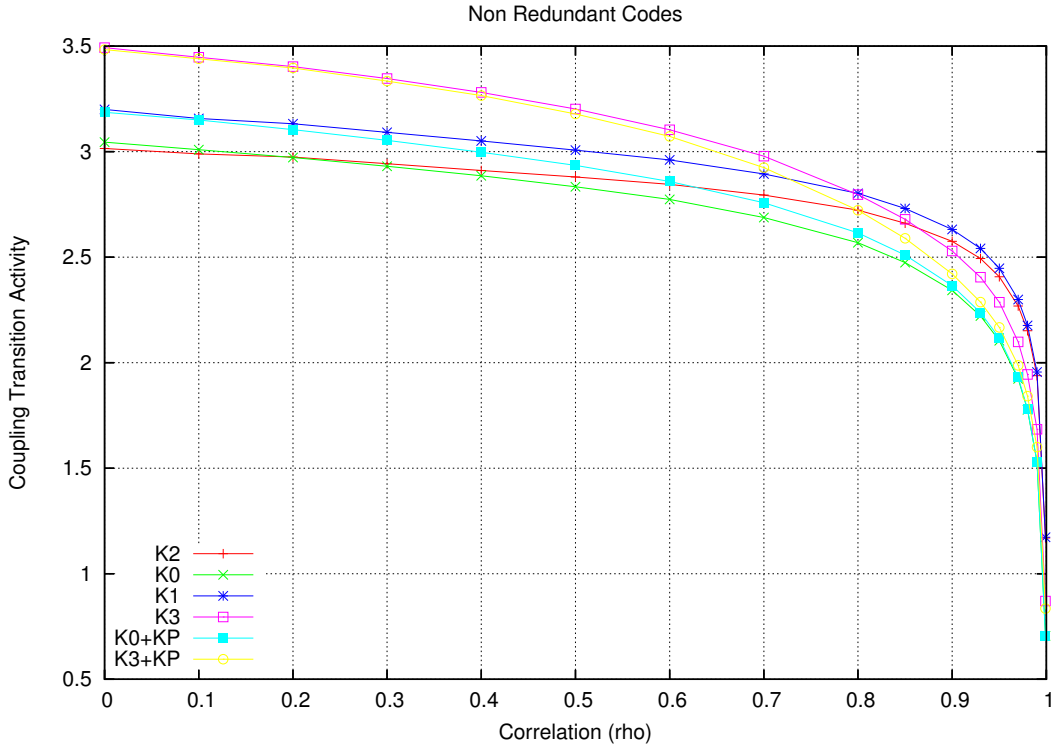


Fig. 5.8: Coupling transition activity for varying ρ, $\sigma_n = 0.15625$, and $B = 8$

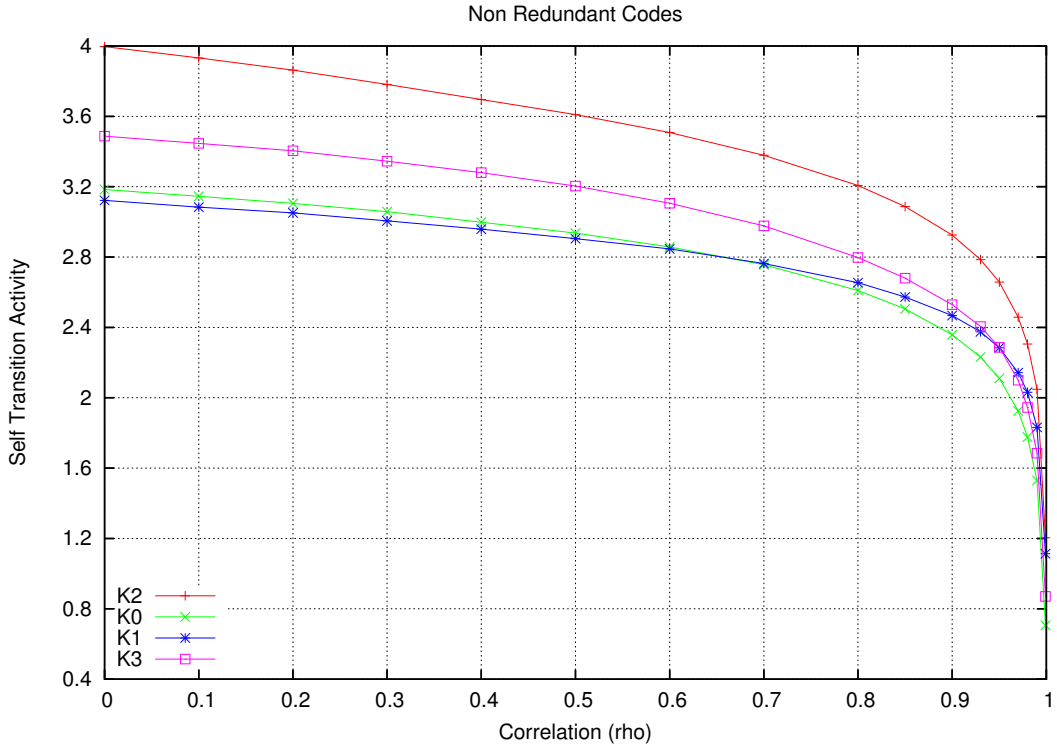


Fig. 5.9: Self transition activity for varying ρ, $\sigma_n = 0.15625$, and $B = 8$

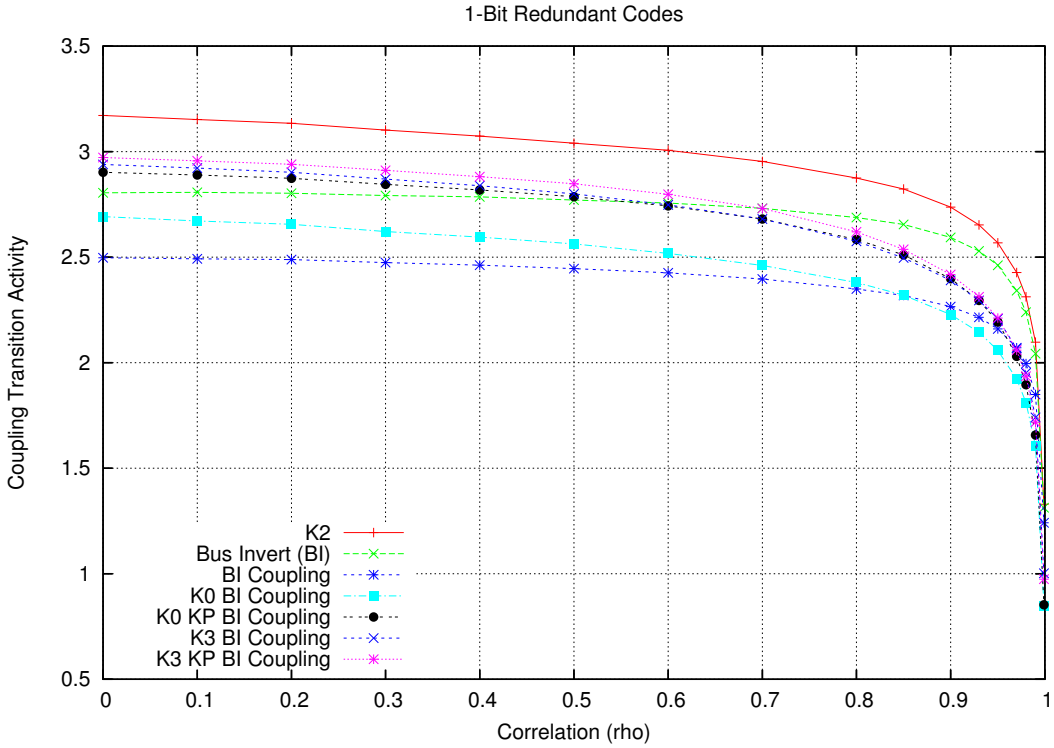


Fig. 5.10: Total coupling transition activity for varying ρ, $\sigma_n = 0.19531$, and $B = 8$

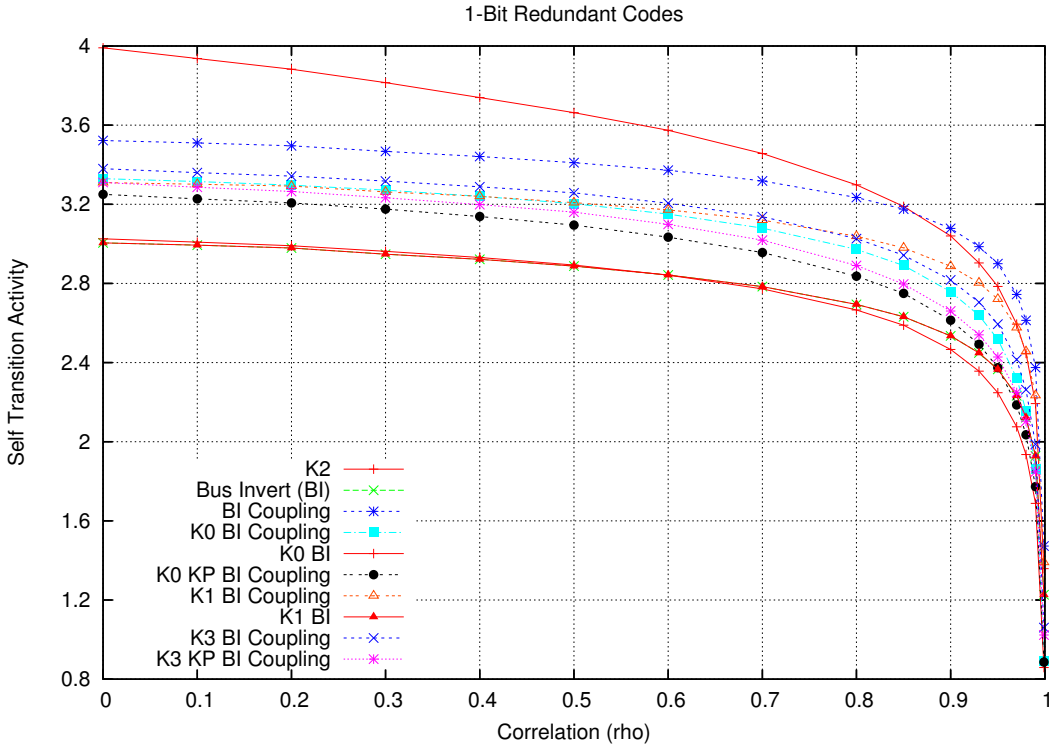


Fig. 5.11: Total self transition activity for varying ρ, $\sigma_n = 0.19531$, and $B = 8$

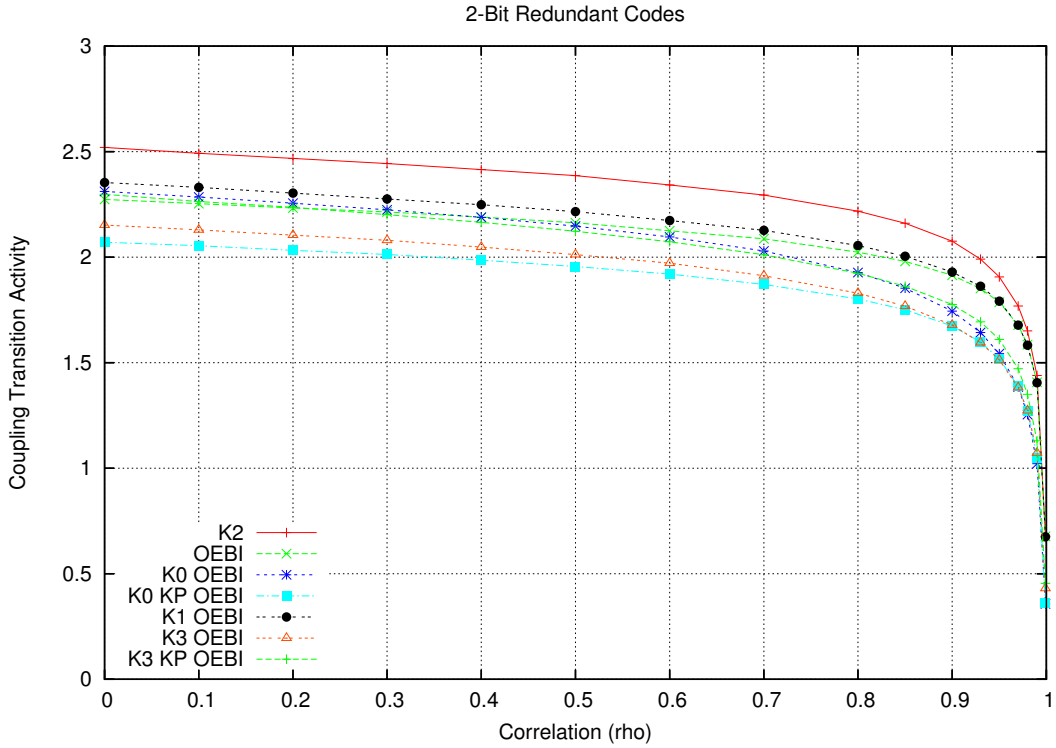


Fig. 5.12: Total coupling transition activity for varying ρ, $\sigma_n = 0.078125$, and $B = 8$

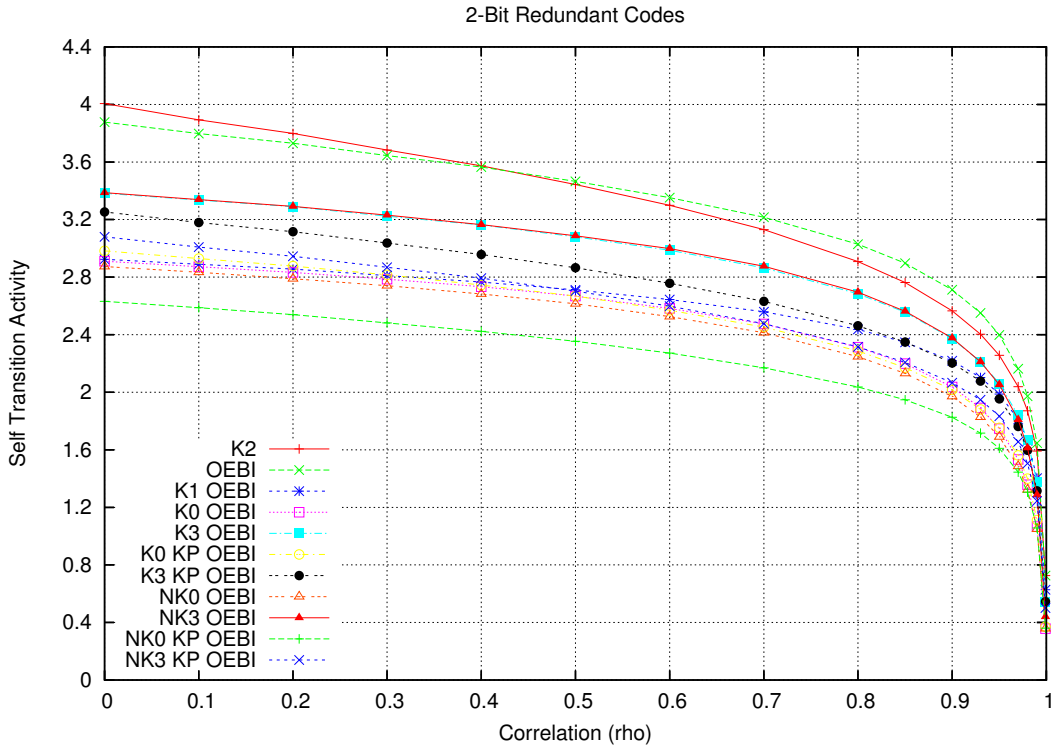


Fig. 5.13: Total self transition activity for varying ρ, $\sigma_n = 0.078125$, and $B = 8$

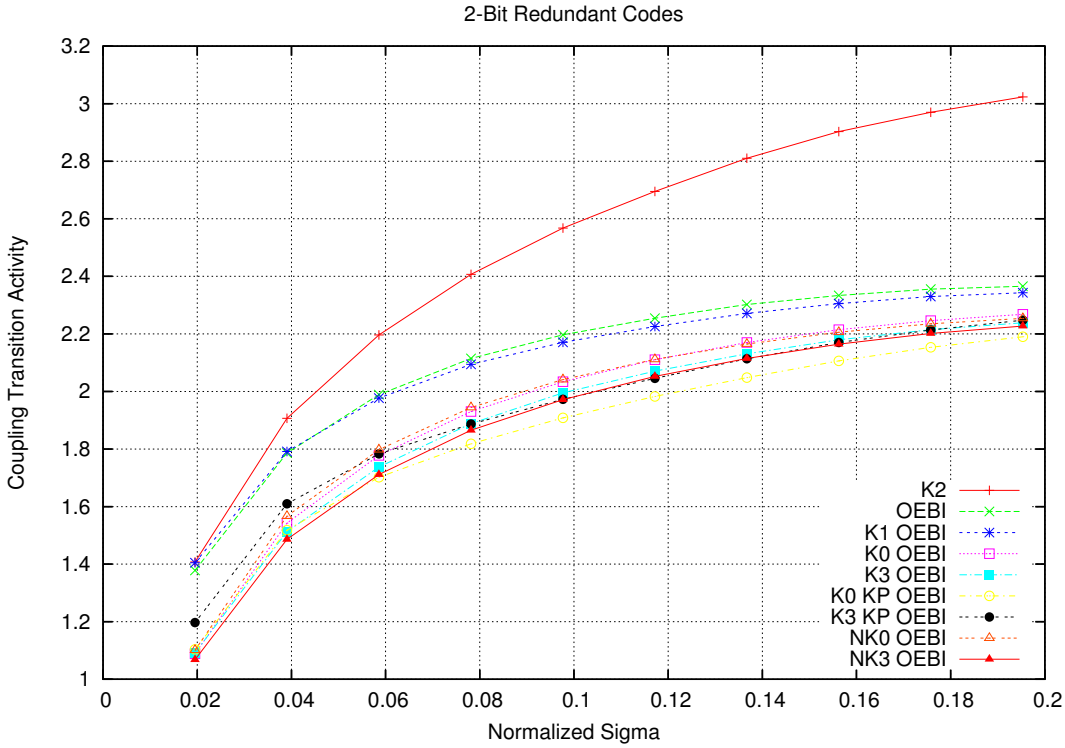


Fig. 5.14: Total coupling transition activity for varying σ_n, $\rho = 0.95$, and $B = 8$

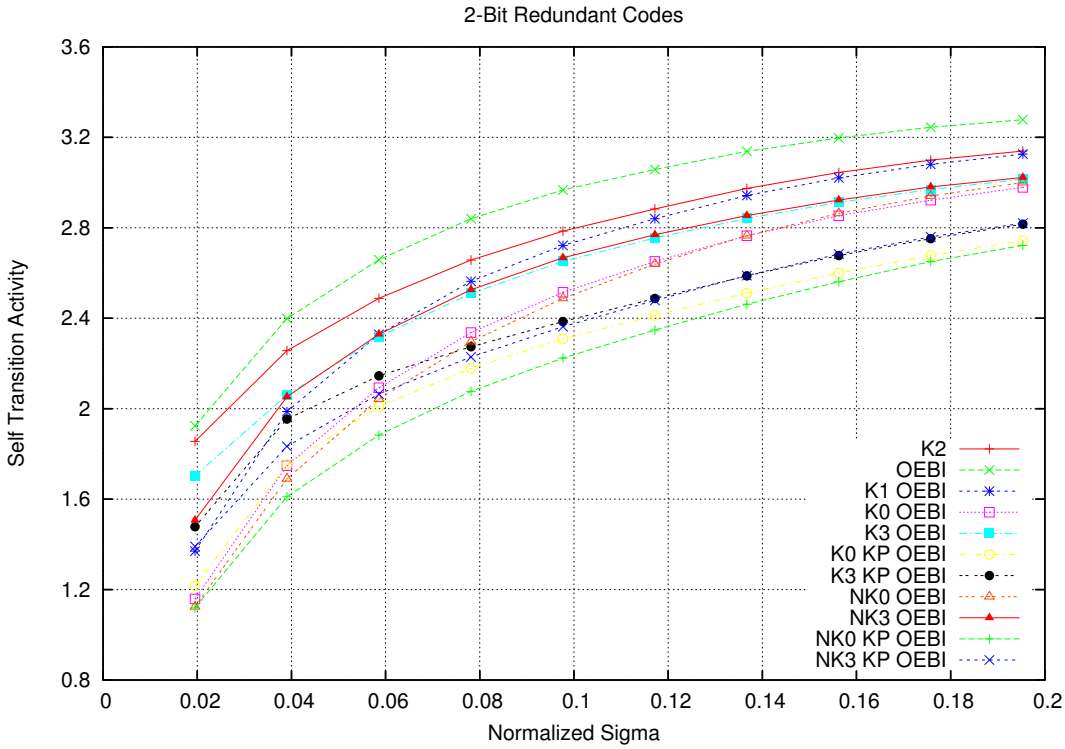


Fig. 5.15: Total self transition activity for varying σ_n, $\rho = 0.95$, and $B = 8$

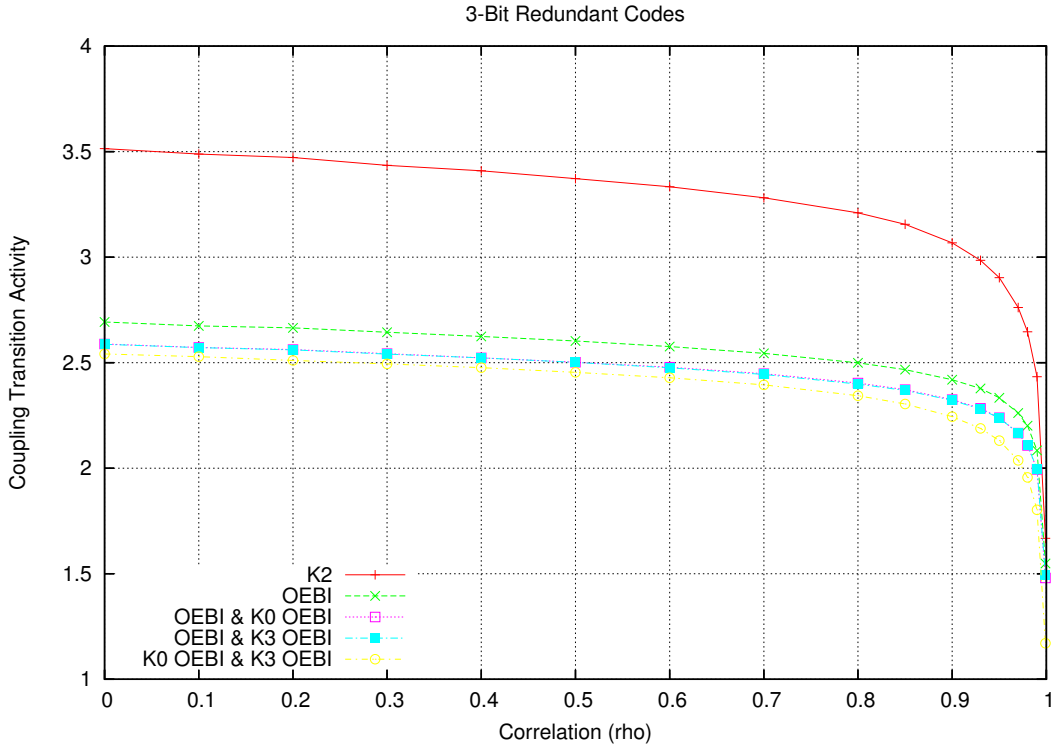


Fig. 5.16: Total coupling transition activity for varying ρ, $\sigma_n = 0.3125$, and $B = 8$

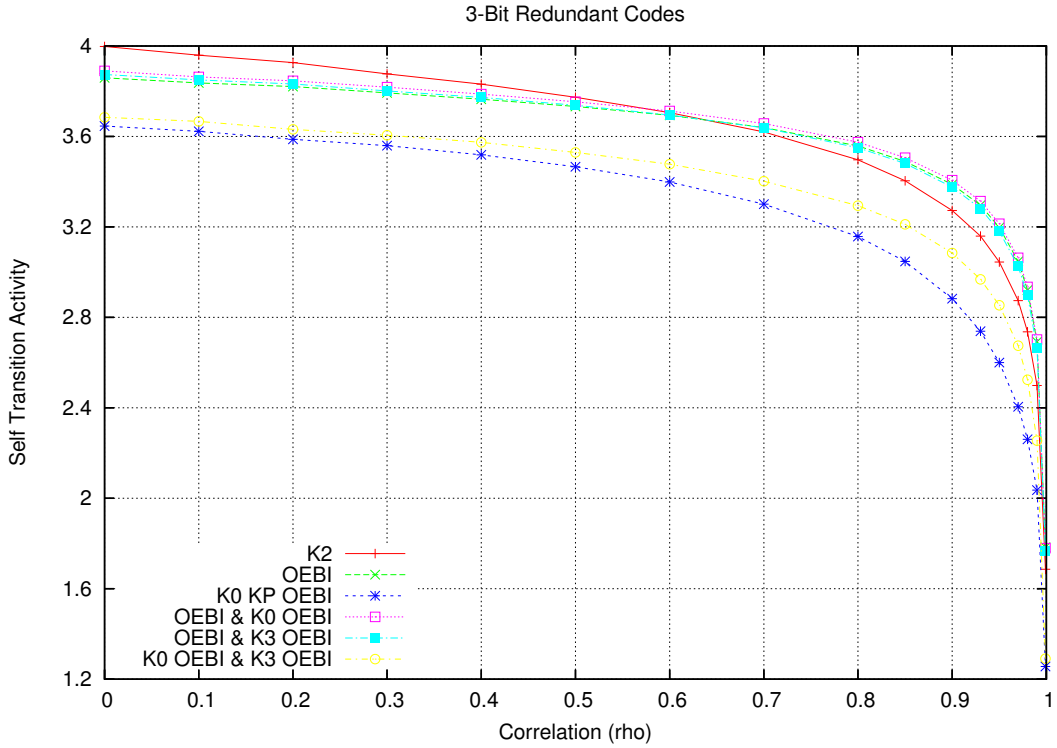


Fig. 5.17: Total self transition activity for varying ρ, $\sigma_n = 0.3125$, and $B = 8$

reduction rather inefficient for DSP signals. It turns actually out, that OEBI is generating a slightly higher self activity than the plain K2 representation. In this case K0-based OEBI schemes outperform OEBI by more than 30%.

Fig. 5.14 and **Fig**. 5.15 analyze the effectiveness of OEBI schemes from another perspective. Here, ρ is held at a constant value and σ_n is varied. For very small values of σ_n, K0-based schemes improve the coupling activity by more than 20%. It can also be observed that OEBI improves and the gap between OEBI and K3-based and K0-based OEBI schemes shrinks as σ_n increases. This can be explained through the fact that for higher σ, there is less spatial correlation for K0 and K3 to take advantage of. Moreover, it is to be noticed also that for very small values of σ_n, K3 behaves in a similar manner as K0. This was expected, as for increasing ρ and decreasing σ_n K3 converges to K0. In terms of self transition activity, we can see that for high correlation factors, OEBI generates a higher activity even than K2, while K0-based OEBI schemes achieve a reduction between 17% and 40%.

Finally, we have also analyzed the chosen 3-bit redundant codes (see **Fig**. 5.14 and **Fig**. 5.15). By adding one redundant line more than OEBI and by combining K0-OEBI with K3-OEBI, we can obtain a slightly higher improvement compared to the case of the previously modified OEBI schemes. Nonetheless, the improvement might be insignificant or even negative due to the fact that one extra line is required, the bus lines becoming thus more closely spaced. As previously mentioned, the main advantage of this scheme is its architectural efficiency in the case of a bidirectional communication bus as K0 is the decoder of K3 and vice-versa (K0 and K3 are so-called dual codes).

It can be concluded that K0 and K3 significantly reduce the coupling and self transition activity especially for high correlations when combined with BI schemes or when used for constructing enhanced BI codes.

In order to determine the efficiency of the analyzed codes when applied in real environments, we have studied the self and the coupling transition activity for a variety of DSP signals. The exact transition activities are determined for the following set of signals:

- s_1 Real part of an FFT output of an IEEE 802.11g OFDM transmitter, using a 64QAM modulation: 700000 samples, B=9, σ_n=0.1072, and ρ=0.271.
- s_2 Imaginary part of an FFT input of an IEEE 802.11g OFDM receiver, using a 64QAM modulation and a type D channel: 700000 samples, B=9, σ_n=0.1118, and ρ=-0.001.
- s_3 Pressure measurement data in a car ignition knock detector: 2300 samples, B=10, σ_n=0.1288, and ρ=0.389.
- s_4 Short piece of classical music (Bach): 1300000 samples, B=$\{8,16\}$, σ_n=0.0448, and ρ=0.984.
- s_5 Short piece of an old piano interpretation (Mozart): 1000000 samples, B=$\{8,16\}$, σ_n= 0.1044, and ρ=0.945.

s_6 Short piece of modern punk rock music (Offspring): 1000000 samples, B={8,16}, σ_n=0.0804, and ρ=0.975.

s_7 Short piece of book reading: 1200000 samples, B={8,16}, σ_n=0.0641, and ρ=0.949.

s_8 Interpretation of a Shakespeare sonnet: 800000 samples, B={8,16}, σ_n=0.0369, and ρ=0.961.

The abovementioned signals are a representative set of DSP signals. The normalized standard deviation varies from 0.0369 to 0.1288 and the correlation factor varies from 0 to almost 0.99. Similarly to the case of synthetic data, the performance of the codes vary significantly for ρ between 0.9 and 0.99. The reason behind choosing more signals in this domain was that typical audio signals are characterized by correlation factors within that interval. Moreover, several audio signals with different σ_n have been selected. The results of the performed simulations are synthesized in **Tab**. 5.1 and **Tab**. 5.2. Note that instead of self transition activity, one can calculate the so-called simplified self transition activity, $T_s' = \mathbf{E}[b_i^+ \Delta b_i] = \frac{T_s}{2}$ as done for **Tab**. 5.2.

The analyses performed on synthetic Gaussian distributed data give us an excellent hint of how the different coding schemes perform on real data. Thus, by using K0 in conjunction with the BI-based schemes, we obtain the most efficient schemes for transition activity reduction. K3-based BI codes are better in terms of self and coupling activity than K0-based ones for signals with a very high temporal correlation and a relatively small σ_n which are represented on wider buses. As expected, OEBI-based schemes exploit both spatial and temporal bit correlation in a more efficient manner than BI-based ones, especially for wider buses, and reduce both self and coupling transition activity.

Bus invert based codes show shortcomings especially for DSP data with a medium to small standard deviation and a high to very high temporal correlation. In general, it can be said that the correlation is a measure of the temporal bit correlation and that the standard deviation is a measure of the spatial correlation in the most significant bits, i.e. for a small σ the spatial correlation in the significant bits is high.

Let us first consider ρ fixed and σ variable. For smaller values of σ, the interval of the random LSBs is reduced and that of the MSBs is increased. The effect is that the total spatial correlation in the MSBs increases and that the total coupling activity is reduced. Moreover, as the LSB interval shrinks and more LSB pass to the middle region and some lines from the middle region to the MSB one, the self activity gets also reduced. Thus, a decreasing standard deviation reduces both coupling and self activities.

Now, let us consider the case of a varying ρ with σ fixed. An increasing ρ does not affect $BP1$ and only $BP0$ decreases slightly, which means an increase in the the spatial correlation between MSBs. Thus the transition activities are reduced by a certain amount especially for narrow buses. Furthermore, when ρ increases, the self transition activity in the MSBs diminishes which implies a smaller total self transition activity. Basically, we can say that σ is a measure of both spatial and temporal correlations and that the

Tab. 5.1: Coupling transition activity

	K2	K1		K0		K3		K0KP		K3KP		BIH		BIC		K0BIC		K3BIC		K0KPBIC		K3KPBIC		OEBI		K0OEBI		K1OEBI		K3OEBI		K0KPOEBI		K3KPOEBI		K0K3OEBI		CK0OEBI		CK3OEBI	
	T_c	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]	T_c	[%]
S_1[9b]	3.61	3.77	4.49	3.64	-0.75	3.90	-8.01	3.40	5.83	3.49	3.26	3.23	10.57	2.91	19.45	3.09	14.36	3.30	8.53	2.92	19.16	3.01	16.46	2.90	19.50	2.81	22.09	2.81	22.08	2.69	25.49	2.74	24.13	2.83	21.51	2.76	23.51	2.74	24.04	2.66	26.30
S_2[9b]	3.61	3.69	-1.98	3.46	4.20	3.77	-4.24	3.00	17.01	3.30	8.65	3.16	12.60	2.80	22.40	2.98	17.58	3.23	10.59	2.74	24.20	2.92	19.28	2.86	20.76	2.72	24.88	2.75	23.90	2.56	29.18	2.57	29.03	2.76	23.72	2.72	24.79	2.67	26.09	2.57	28.92
S_3[10b]	4.84	4.96	-2.43	2.25	53.47	3.41	29.67	3.83	21.01	3.36	30.57	3.01	37.87	2.92	39.77	2.12	56.27	2.76	43.10	2.99	38.23	2.71	44.09	1.92	60.32	2.19	54.82	2.08	57.08	2.52	47.97	2.41	50.32	2.63	45.72	1.93	60.23	1.93	60.14	2.19	54.82
S_4[8b]	1.27	1.28	-0.98	0.93	27.01	1.10	13.27	0.92	27.29	1.01	20.10	1.26	0.64	1.10	12.94	0.89	29.95	1.06	16.71	0.91	28.48	1.00	20.93	1.22	3.73	0.90	28.68	1.22	3.83	0.95	25.32	0.91	27.96	0.99	21.97	1.18	6.66	1.19	6.51	0.89	29.73
S_4[16b]	1.27	1.32	-4.34	1.36	-7.24	1.10	13.27	1.36	-7.28	3.64	-187.1	1.27	-0.00	1.25	1.81	1.35	-6.03	1.06	16.71	1.36	-6.95	3.13	-146.8	1.25	1.81	1.35	-6.03	1.30	-2.54	1.09	13.73	1.36	-7.20	1.52	-19.58	1.22	3.90	1.22	4.06	1.30	-2.22
S_5[8b]	2.45	2.49	-1.65	2.15	12.21	2.31	5.62	2.16	11.63	2.21	9.78	2.35	3.88	2.07	15.21	1.96	19.82	2.12	13.52	2.08	15.05	2.12	13.50	2.12	13.52	1.93	20.99	2.09	14.44	1.91	22.12	1.84	24.83	1.90	22.20	2.03	17.09	2.02	17.26	1.87	23.70
S_5[16b]	5.13	4.99	2.79	4.73	7.83	5.79	-12.81	4.59	10.51	5.96	-16.25	4.74	7.59	4.46	13.02	4.44	13.43	5.38	-4.91	4.41	13.96	5.76	-12.40	4.06	20.76	4.07	20.72	4.00	21.93	4.63	9.69	4.26	17.00	4.43	13.65	3.86	24.67	3.93	23.42	3.95	22.92
S_6[8b]	1.71	1.74	-1.45	1.39	18.73	1.57	8.24	1.39	18.65	1.47	13.97	1.68	1.74	1.49	12.89	1.30	23.75	1.48	13.76	1.36	20.63	1.44	15.94	1.57	8.11	1.32	22.83	1.57	8.51	1.33	22.21	1.29	24.53	1.37	19.87	1.51	11.73	1.51	11.74	1.29	24.69
S_6[16b]	5.72	5.75	-0.45	5.36	6.38	5.56	2.84	5.36	6.29	5.54	3.14	5.45	4.82	5.02	12.33	4.94	13.72	5.14	10.17	5.19	9.30	5.30	7.42	4.86	15.09	4.70	17.87	4.85	15.24	4.67	18.49	4.48	21.66	4.58	19.99	4.59	19.79	4.54	20.73	4.44	22.38
S_7[8b]	1.05	1.07	-1.82	0.86	17.63	0.99	5.78	0.87	17.03	0.93	11.19	1.02	2.27	0.90	13.70	0.80	23.92	0.92	12.61	0.84	19.90	0.89	14.68	0.95	8.85	0.81	23.11	0.95	9.17	0.81	22.27	0.78	25.07	0.84	19.84	0.92	12.35	0.92	12.17	0.79	24.98
S_7[16b]	3.58	3.61	-0.62	3.37	5.95	3.51	2.04	3.38	5.71	3.49	2.55	3.41	4.90	3.14	12.43	3.09	13.73	3.23	9.87	3.26	9.17	3.32	7.34	3.05	14.89	2.95	17.71	3.05	14.98	2.93	18.29	2.81	21.50	2.88	19.71	2.88	19.58	2.85	20.53	2.79	22.24
S_8[8b]	1.04	1.05	-1.59	0.80	22.95	0.96	7.71	0.80	22.87	0.87	16.09	1.04	0.01	0.84	18.83	0.75	27.81	0.90	12.86	0.78	24.76	0.85	17.60	0.99	4.65	0.77	25.48	0.99	4.06	0.78	24.34	0.77	25.79	0.83	19.58	0.96	7.67	0.96	7.70	0.76	26.87
S_8[16b]	2.23	2.30	-3.03	2.01	9.76	2.91	-30.52	1.98	11.41	3.48	-56.20	2.14	3.88	1.96	12.05	1.96	12.24	2.66	-19.18	1.94	12.85	3.43	-53.78	2.05	7.89	1.93	13.60	2.14	4.00	2.34	-4.90	1.94	12.98	2.16	3.00	1.89	15.19	1.90	14.69	1.89	15.20

Tab. 5.2: Simplified self transition activity

	K2		K1		K0		K3		K0KP		K3KP		BIH		K0BIC		K3BIC		K0KPBIC		K3KPBIC		OEBI		K0OEBI		K1OEBI		K3OEBI		K0KPOEBI		K3KPOEBI		K0K3OEBI		CK0OEBI		CK3OEBI	
	T_s'	T_s' [%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]	T_s'	[%]
$\mathcal{S}_1$[9b]	2.18	1.87	13.93	1.91	12.51	1.98	9.11	1.91	12.51	1.98	9.1	1.75	19.60	1.98	9.27	1.89	13.31	1.91	12.25	1.75	19.42	1.80	17.34	2.12	2.72	1.99	8.81	1.98	9.26	2.06	5.59	1.84	15.44	1.88	13.59	2.16	0.68	2.17	0.48	2.09	4.13
$\mathcal{S}_2$[9b]	2.26	1.78	21.33	1.75	22.32	1.92	15.03	1.75	22.32	1.92	26.25	1.66	26.30	1.95	13.61	1.80	20.23	1.85	17.84	1.56	30.75	1.72	23.96	2.13	5.76	1.86	17.56	1.90	15.90	1.97	12.79	1.65	27.01	1.78	21.15	2.16	4.29	2.17	4.01	1.98	12.21
$\mathcal{S}_3$[10b]	2.21	2.04	7.69	2.01	9.27	2.07	6.31	2.01	9.27	2.07	6.31	1.67	24.46	1.79	19.13	1.84	16.96	1.79	19.13	1.69	23.67	1.84	16.96	2.44	-10.06	1.97	11.24	2.34	-5.92	1.87	15.38	1.55	29.78	2.01	9.27	2.42	-9.47	2.43	-9.86	1.97	11.24
$\mathcal{S}_4$[8b]	0.72	0.59	18.78	0.46	35.87	0.55	23.65	0.46	35.87	0.55	23.65	0.58	19.85	0.86	-18.52	0.52	28.48	0.60	17.12	0.49	32.58	0.56	23.25	0.75	-3.92	0.49	32.02	0.63	13.43	0.65	10.17	0.50	31.20	0.59	18.92	0.76	-5.17	0.75	-3.11	0.50	30.88
$\mathcal{S}_4$[16b]	0.72	0.77	-6.84	0.68	5.51	2.07	-185	0.68	5.51	2.07	-185	0.72	-0.00	0.79	-9.14	0.72	0.20	2.02	-178.4	0.70	3.47	1.85	-156	0.80	-10.14	0.73	-0.35	0.83	-15.22	2.07	-185.4	0.71	2.27	0.85	-17.26	0.82	-13.50	0.80	-10.1	0.77	-6.54
$\mathcal{S}_5$[8b]	1.35	1.17	13.15	1.09	19.38	1.17	13.45	1.09	19.38	1.17	13.45	1.13	16.00	1.43	-5.77	1.20	10.63	1.25	7.32	1.14	15.72	1.17	13.10	1.43	-6.22	1.20	11.26	1.30	3.47	1.29	4.30	1.11	17.56	1.16	13.59	1.44	-6.71	1.41	-4.89	1.20	10.64
$\mathcal{S}_5$[16b]	2.70	2.39	11.48	2.38	11.95	2.97	-9.82	2.38	11.95	2.97	-9.82	2.30	14.98	2.54	6.01	2.46	9.08	3.08	-13.84	2.43	10.22	2.91	-7.89	2.68	0.86	2.54	6.08	2.55	5.79	2.99	-10.63	2.64	2.31	2.87	-6.06	2.76	-2.12	2.76	-2.14	2.64	2.37
$\mathcal{S}_6$[8b]	0.97	0.82	16.00	0.70	27.55	0.79	18.27	0.70	27.55	0.79	18.27	0.80	18.09	1.08	-10.74	0.79	18.63	0.86	11.28	0.74	23.72	0.80	17.85	1.03	-5.93	0.77	20.93	0.90	7.69	0.90	7.01	0.74	23.92	0.82	15.72	1.03	-6.48	1.01	-4.05	0.78	20.00
$\mathcal{S}_6$[16b]	2.98	2.81	5.68	2.68	10.06	2.78	6.75	2.68	10.06	2.78	6.75	2.73	8.39	3.08	-3.22	2.88	3.50	2.91	2.31	2.72	8.66	2.78	6.72	3.12	-4.78	2.90	2.59	3.03	-1.65	2.97	0.39	2.63	11.75	2.69	9.65	3.13	-5.10	3.10	-3.98	2.94	1.55
$\mathcal{S}_7$[8b]	0.61	0.50	17.42	0.44	28.15	0.50	18.35	0.44	28.15	0.50	18.35	0.49	19.75	0.66	-8.48	0.49	20.35	0.53	12.69	0.46	24.83	0.50	18.34	0.64	-4.53	0.48	22.08	0.55	9.88	0.57	7.49	0.46	24.69	0.51	16.05	0.64	-4.94	0.63	-3.18	0.48	21.28
$\mathcal{S}_7$[16b]	1.88	1.77	6.17	1.69	10.33	1.76	6.83	1.69	10.33	1.76	6.83	1.71	9.00	1.94	-2.98	1.81	4.15	1.83	2.71	1.71	9.29	1.75	7.30	1.97	-4.32	1.82	3.37	1.90	-0.95	1.87	0.88	1.65	12.29	1.69	10.06	1.97	-4.65	1.95	-3.55	1.84	2.15
$\mathcal{S}_8$[8b]	0.63	0.48	24.38	0.40	36.08	0.49	23.30	0.40	36.08	0.49	23.30	0.47	25.12	0.82	-28.81	0.47	25.09	0.54	14.24	0.43	31.72	0.49	22.80	0.66	-4.14	0.43	32.03	0.51	18.67	0.59	7.05	0.43	32.02	0.51	19.24	0.66	-4.99	0.65	-2.08	0.43	31.31
$\mathcal{S}_8$[16b]	1.35	1.23	8.95	1.00	26.07	1.69	-24.8	1.00	26.07	1.69	-24.8	1.19	12.21	1.49	-10.38	1.04	23.26	1.76	-30.33	1.04	23.27	1.64	-21.64	1.36	-0.61	1.06	21.62	1.28	4.97	1.66	-22.7	1.06	21.40	1.68	-24.20	1.47	-8.56	1.48	-9.34	1.16	14.32

correlation factor ρ is virtually not giving any measure of the spatial correlation in the case of wider buses.

The classic BI schemes perform poorly for high values of ρ and small values of σ. However, exactly those ranges of values appear in typical DSP applications. Among non-redundant schemes, the high potential in terms of transition activity reduction present in those situations is perfectly exploited by K0-based schemes. K3 tries to make use of the spatial correlation when spread over the complete width of the bus, and this turns to be an advantage only for wide buses, small standard deviation and high correlation factors, i.e. a very strong spatial correlation. Nevertheless, the enhanced OEBI code in which K3 has been combined with K0 proved to be a very efficient, robust, and flexible coding scheme for coupling transition activity reduction despite the extra redundancy bit compared to OEBI.

5.4 Low Complexity Partial Bus Invert Coding

As shown in **Sec**. 5.1, there is a high temporal and spatial correlation in the most significant bits in the case of DSP signals, which implies a low self and coupling activity in those bits.

5.4.1 Partial BI and OEBI for DSP Signals

An efficient low-power code should not destroy the spatial and temporal correlation in the highly correlated bits and focus on reducing the activity in those which are lowly correlated. Actually, this is rather easy to achieve in the case of DSP signals. Let us consider the case of partial bus-invert (PBI) which has been proposed to meet the requirement of application-specified systems and avoid unnecessary inversions in highly correlated bits (see [171]). Basically, we have just to estimate the regions of the MSBs and that of the LSBs. Once the regions have been identified, BI can be applied only to the LSBs. In this way, PBI becomes very simple in the case of DSP signals.

We have implemented PBI for self activity and adapted it for coupling activity. For instance, PBIH4 means that the classic BI based on the Hamming distance (hence BIH) has been applied up to the fourth bit. PBIC means that the Hamming distance has been replaced with a coupling metric in order to reduce coupling activity, and finally, K0PBI denotes the conjunction of the non-redundant K0 code with a PBI scheme.

One essential contribution of this chapter is the development of the Partial OEBI (POEBI) scheme for DSP signals. POEBI is constructed in a similar fashion as PBIC. Basically, the coupling activity is reduced only in the LSBs by applying OEBI. Compared to PBI, POEBI is more flexible and due to these higher degrees of freedom, we can expect it to be more efficient.

In order to analyze the effectiveness of the proposed encoding schemes, we have performed extensive simulations with large sets of synthetic and real data from various DSP applications. Like explained in [145], signals obtained from sources such as speech, audio, and video can be modeled using ARMA (Auto-Regressive Moving Average) models. Therefore, we have employed these ARMA models to generate zero-mean Gaussian distributed data with varying standard deviation and correlation factor. The coupling and self transition activity for synthetic data represented on eight bits are illustrated in **Fig**. 5.18 and **Fig**. 5.19. We first discuss the behavior of the PBI-based schemes and afterwards that of the POEBI-based ones.

With regards to T_s, BIH an K0BIH behave best at very small values of ρ and become less efficient when ρ increases. This can be explained as follows. For $\rho = 0$, we have $t_m = 0.5$, which actually means that there is no breakpoint and thus, no temporal correlation to take advantage of. For high ρ-s, K0BIH slightly improves the temporal correlation. However, the improvement is rather insignificant (less than 2%) because the temporal correlation in the MSBs is poorly exploited. On the contrary, the PBI-based schemes are able to efficiently exploit the high temporal and spatial correlation for high ρ-s. As expected, in terms of reducing T_s, the PBIH schemes perform better than the PBIC ones and KOPBI manage to achieve better results than K2PBI. Similarly, the codes tailored for reducing T_s behave better than those designed for T_s optimization. Moreover, K0 helps the encoding schemes to better exploit the spatial and temporal correlation. In the represented case, K2BIC and K2PBIC4 reduce T_c by around 14% and 19%, respectively. With the proposed K0PBIC4, we achieve a reduction of more than 27%.

Even more interesting results are obtained when applying OEBI and POEBI. We analyze first the results from the perspective of T_s. It can be observed that OEBI behaves poorly for DSP signals, especially for high values of ρ. Nevertheless, POEBI schemes behave better than PBIC. The reason behind that is that in order to reduce the coupling activity, POEBI inverts fewer bits than PBI, affecting thus less the self activity. Furthermore, K0POEBI4 is basically as efficient as the best BI-based scheme. For ρ greater than 0.96, K0POEBI3 is also very effective. Consequently, we can say that POEBIs significantly reduce T_s.

Nonetheless, OEBI has been developed for reducing mainly the total coupling activity, T_c. In **Fig**. 5.18 and **Fig**. 5.19, we can see that the POEBI-based schemes dramatically improve the efficiency of the classic OEBI scheme for any ρ and clearly outperform the more simple PBIs, although they introduce a supplementary redundancy bit. Let us first consider the case of highly correlated data, f.i. $\rho = 0.95$. The classic OEBI reduces T_c by 19% and K0PBIC4 (the best PBIC) by more than 27%. In contrast, the POEBI-based schemes achieve improvements of more than 40%, with K0POEBI3 and K0POEBI4 reaching almost 45%. The amelioration is also remarkable for poorly correlated signals. For example, when $\rho = 0.4$, K0POEBI4 reduces T_c by more than 35% while OEBI and K0PBIC achieve reductions of around 23% and 17%, respectively. It is to be noticed that K0 allows OEBI to exploit more efficiently the spatial correlation in the MSBs.

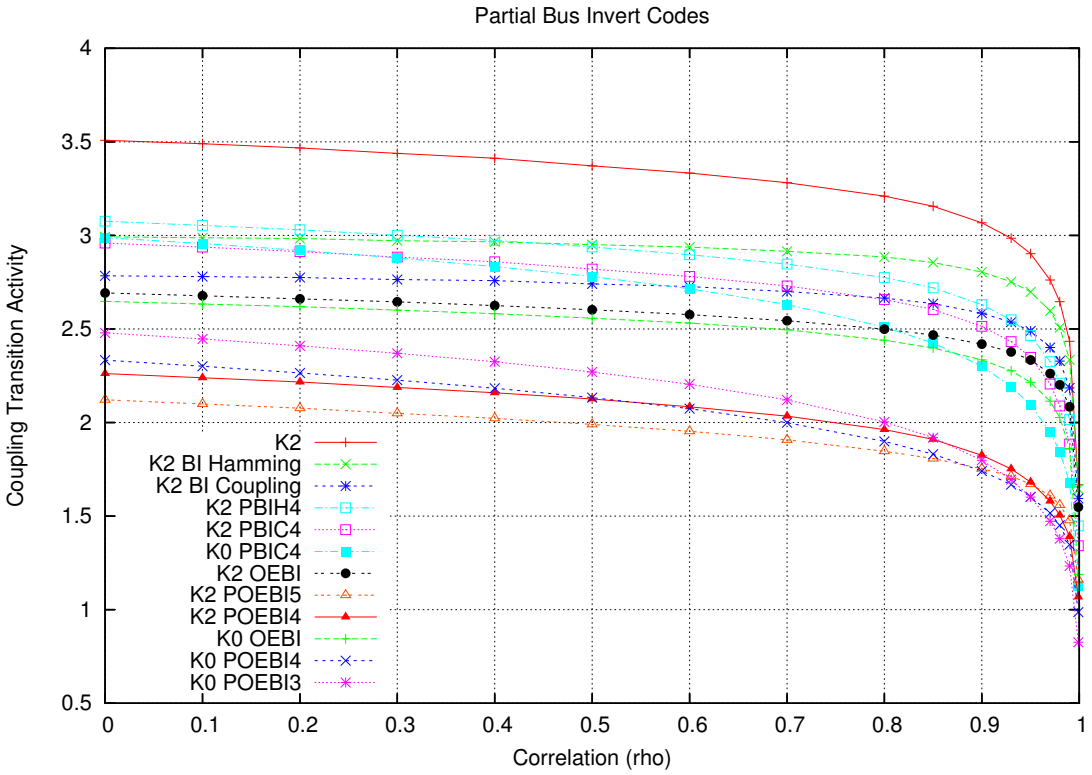


Fig. 5.18: Total self transition activity for fixed σ and varying ρ

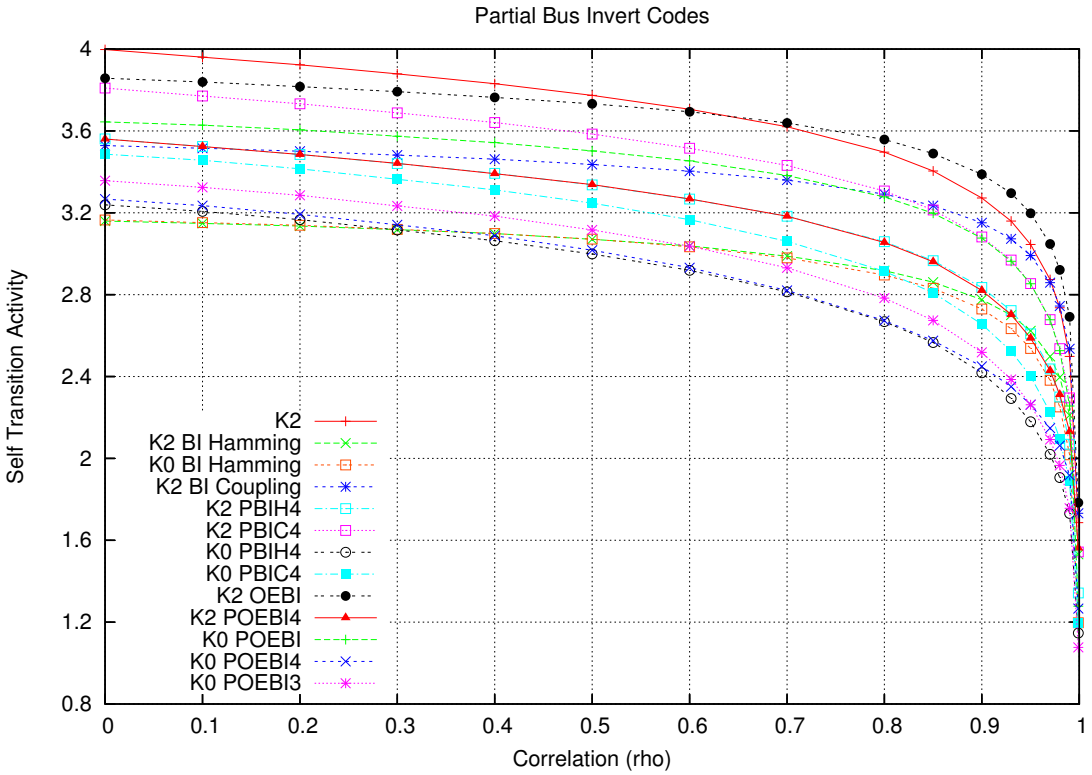


Fig. 5.19: Total self transition activity for fixed σ and varying ρ

Tab. 5.3: Self and coupling transition activities for real DSP data

	K2	*BIH*	*BIC*	*OEBI*	*PBIH*	*PBIC*		*POEBI*	
s_1					*K0PBIH6*	*PBIC6*	*K0PBIC5*	*POEBI5*	*K0POEBI5*
T_c	3.60	3.22	2.91	2.90	3.16	2.89	2.99	2.22	2.34
T_s	4.34	3.50	3.94	4.22	3.20	4.04	3.54	3.88	3.38
s_2					*K0PBIH4*	*PBIC6*	*K0PBIC5*	*POEBI5*	*K0POEBI4*
T_c	3.61	3.14	2.78	2.86	2.99	2.89	2.83	2.22	2.26
T_s	4.50	3.32	3.88	4.24	3.00	4.20	3.22	4.06	3.06
s_3					*K0PBIH6*	*PBIC8*	*K0PBIC8*	*POEBI7*	*K0POEBI3*
T_c	4.84	2.86	2.83	1.92	2.17	2.77	1.85	1.24	1.81
T_s	4.42	3.34	3.56	4.86	2.28	3.58	3.70	4.10	3.94
s_4					*K0PBIH2*	*PBIC2*	*K0PBIC2*	*POEBI2*	*K0POEBI2*
T_c	1.26	1.25	1.15	1.22	0.86	0.97	0.81	0.77	0.68
T_s	1.44	1.16	1.62	1.50	0.80	1.36	0.90	1.22	0.86
s_6					*K0PBIH2*	*PBIC4*	*K0PBIC3*	*POEBI3*	*K0POEBI3*
T_c	1.71	1.67	1.53	1.57	1.21	1.36	1.18	1.00	0.92
T_s	1.94	1.58	2.04	2.04	1.22	1.80	1.32	1.70	1.26
s_8					*K0PBIH2*	*PBIC2*	*K0PBIC2*	*POEBI2*	*K0POEBI2*
T_c	1.03	1.03	0.95	0.98	0.71	0.79	0.70	0.65	0.60
T_s	1.26	0.94	1.36	1.32	0.70	1.20	0.78	1.12	0.74

The abovementioned comments can be resumed as follows. The MSBs are spatially and temporally highly correlated while the LSBs are uncorrelated and uniformly distributed. The non-redundant K0 code efficiently exploits the spatial correlation existing in the MSBs of typical DSP signals reducing thus the self activity, especially for smaller values of ρ. The temporal correlation is preserved and in some cases (depending on σ and ρ) even slightly reduced. The PBI-based schemes exploit than this temporal correlation by applying BI only on the LSBs. As these bits are uncorrelated and uniformly distributed, BI behaves in the case of self-activity optimally among 1-bit redundant low-power codes [187]. Similarly, POEBI is taking advantage of the high temporal and spatial correlation in the MSBs by not destroying it. Due to the extra bit, these schemes have more degrees of freedom than the PBIs. In contrast to the PBIs, in the case of the POEBI codes, the LSBs can be inverted completely or only partially. This offers higher optimization capabilities. Moreover, as the the lower and upper breakpoints for T_s and T_c are practically the same, the POEBI schemes are in general as efficient as the PBI codes, especially when the breakpoints have higher values. In this case, there are more LSBs and the negative effect of having an extra-bit is decreased.

Tab. 5.3 shows the transition activity for BI and OEBI as well as for the best PBI and POEBI. The coding scheme for which the optimum has been achieved is also indicated.

Tab. 5.4: Analysis of total equivalent transition activity for different bus types

	T_c	T_s	$T'_{eq} = T'_s + \kappa T_c$		
			κ=2.76	κ=2.09	κ=0.65
K2	6.01	3.17	19.78	15.74	7.08
K2 BIH	5.69	2.87	18.60	14.78	6.58
K2 BIC	5.17	3.23	17.51	14.04	6.60
K2 PBIH10	5.16	2.74	16.99	13.53	6.10
K2 PBIC11	4.86	2.87	16.30	13.04	6.03
K2 OEBI	5.05	3.30	17.24	13.86	6.58
K2 POEBI10	3.79	2.75	13.24	10.69	5.22
K0	5.72	2.86	18.65	14.81	6.57
K0 BIH	5.53	2.81	18.09	14.38	6.41
K0 BIC	5.24	3.02	17.49	13.98	6.43
K0 PBIH10	4.86	2.43	15.87	12.61	5.60
K0 PBIC10	4.64	2.58	15.41	12.30	5.60
K0 OEBI	4.94	3.10	16.75	13.44	6.31
K0 POEBI9	3.63	2.47	12.50	10.06	4.83

In the case of typical audio signals, i.e. s_4, v_2, and v_3, we can see that K0POEBI is improving the self activity by almost the same amount as the best PBIH. For instance, T_s of v_3 is reduced with K0PBIH2 and with K0POEBI2 by 44.44% and 41.27%, respectively. Further, K0POEBI is more effective in reducing T_c than any other scheme. For example, in the case of v_1, K0POEBI2 reduces T_c by 46.03%, while OEBI and BIC only by 3.17% and 8.73%, respectively. On the contrary, K0PBIC2 significantly reduces T_c by 35.71%. It is worth mentioning, that while OEBI generally reduces T_c by merely 5%, POEBI manages to improve the coupling activity by more than 40%.

Things are similar in the case of s_1, and s_2. There are only two differences mainly because the correlation in OFDM communication systems is close to zero and thus, the breakpoints have higher values. On the one hand, there is less temporal correlation in the MSBs to take advantage and the achieved improvement is slightly smaller than in the previous cases. On the other hand, POEBI and PBI use more bits for coding. The results confirm those obtained with synthetic data.

In the case of signal s_3, which is poorly correlated and is characterized by a large σ_n, there is a behavioral discrepancy between the coding schemes that achieve the best results for T_c and T_s, respectively. While K0PBIH6 reduces T_s by 48.41%, the best OEBI brings an improvement of 10.86%. However, POEBI7 manages to reduce T_c by an incredible 73.38% while K0PBIH6 improves the coupling activity by 55.15%. This shows that, in order to choose the optimal low-power code, designers need in the early stages of the design flow information about the statistical characteristics of the data.

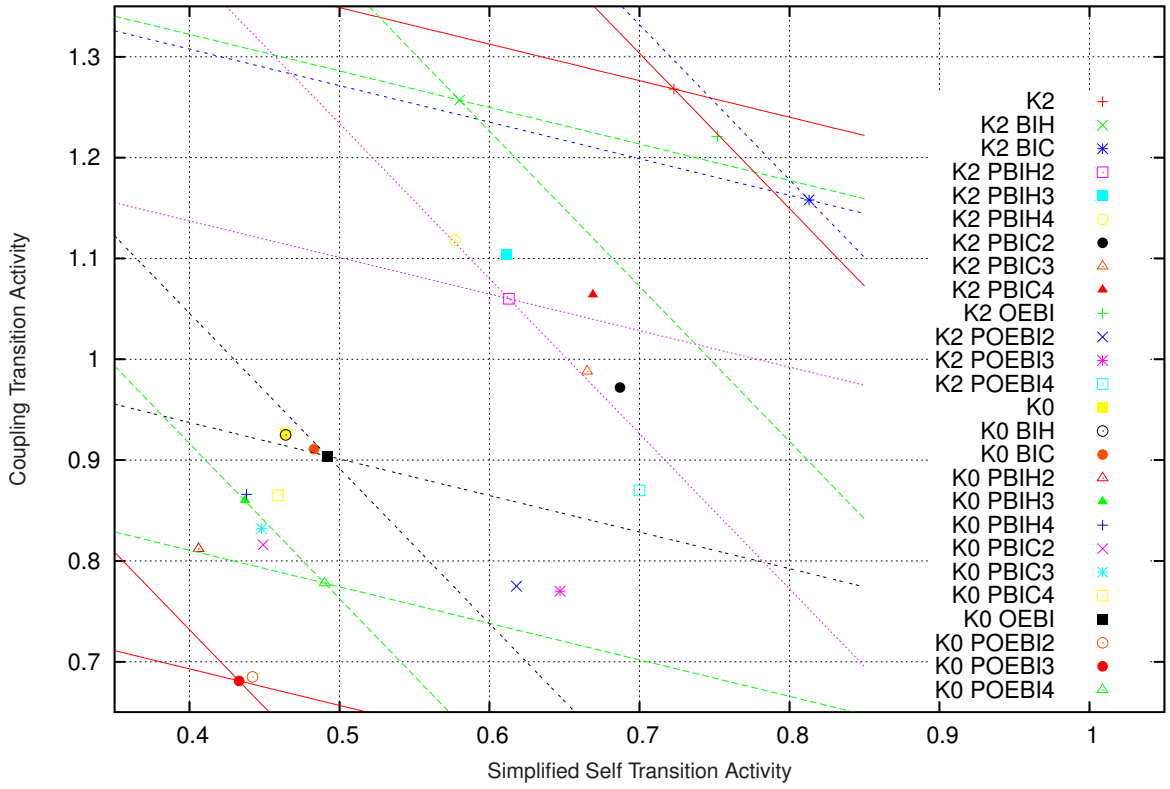


Fig. 5.20: Graphical figure of merit for low-power codes as a function of κ

Tab. 5.4 shows the effectiveness of different codes in a projected 65 nm technological node for v_1 when applied in typical local (κ=2.76), intermediate (κ=2.09), and global wires (κ=0.65). Capacitances and bus aspect factor have been computed by employing the formulas given in **Chap**. 2. As shown in **Fig**. 5.20, by representing each encoding scheme as a point in the $\{T_s',T_c\}$ plane, we can easily define a graphical figure of merit. For a given κ, we draw for each (T_s',T_c) pair the lines defined by $y=\frac{T_s'+\kappa T_c-x}{\kappa}$. The best code is determined by finding the associated line that is closest to the origin.

5.4.2 Efficient Adaptive Partial Bus Invert Coding for DSP Signals

As mentioned in **Chap**. 3, adaptive coding schemes suit their functionality on the statistical characteristics of the transmitted data. The encoder gathers information during the observation window and based on the results, it applies a coding scheme or another. The primary disadvantage of mainly all adaptive schemes is their intrinsic complexity. They generally require large and power-hungry tables, which makes them unsuitable for many classes of applications. In order to avoid such complexity, application-specific coding schemes can be developed.

In typical DSP signals, due to the fact that bus invert is the most efficient among the 1-bit redundant coding schemes [187, 189], it can be applied only to the bits belonging to

the LSB region or to the bits with a self activity higher than an optimally chosen threshold. Thus, when the data characteristics are unknown at design time, the breakpoints have to be estimated on-the-fly before applying BI-based encoding schemes. Finding the breakpoint is equivalent to constructing the PBI mask as described in [87]. The mask does not have to be however calculated by observing each bus line, but by identifying the breakpoints.

With this knowledge, one can envisage several adaptive schemes. Basically, two pointers are required to indicate BP_0 and BP_1. For every observation window, the self activity in the two bits is monitored and based on the result, a control unit keep the pointers unchanged or moves them towards the MSB or the LSB. Thus, only two counters are necessary, which means a significant reduction in comparison with the classical APBI. Moreover, an extra control line is added that signalizes the beginning of a new application window, the position of the breakpoint, and also the applied scheme.

Given that the previously introduced PBIH, PBIC, and POEBI encoding schemes are defined by the bit that is closest to the MSB, the scheme can be further simplified by monitoring only one (generic) breakpoint. That breakpoint can be adjusted to the right or to the left after every observation window. For quasi-stationary data, trimming the pointer by only one bit is effective enough. However, if the statistical characteristics of the data change rather quickly, the breakpoint must also be adapted more rapidly. Two versions of the scheme are imaginable: on the one hand, the pointer can be adjusted with more than one bit, and on the other one, more bits can be monitored in order to keep track of the breakpoints in real time.

Fig. 5.21 and **Fig**. 5.22 illustrate the generic schemes of the proposed Adaptive PBI (APBIH and APBIC) and Adaptive POEBI (APOEBI). The partially inverted values of the bus are compared with the previously sent data for the Hamming or the coupling distance. The most important unit is the controller that monitors the pointers, resets the counters, and generates the masks. Furthermore, the controller can be extended to choose among several coding schemes. This adds however to the complexity and such as scheme remains applicable only if the power consumption on the bus is large enough to come up for extra required energy consumption. If needed, the *Control Unit* can also be configured at run-time by changing the configuration data (*Config Data*).

Simulations have showed that for quasi-stationary signals, the achieved reduction in transition activity is comparable to the one obtained with the static versions of the codings, i.e. from about 20% to 45%. The efficiency of the schemes is given for a large span of observation and application windows length. Nonetheless, for non-stationary signals, the efficiency of the adaptive schemes decreases slightly, that is from less than 12% to around 34%. Furthermore, the efficiency is as expected much more dependent on the lengths of the observation and application windows and also on the number of monitorizing pointers. The existence of two or three pointers significantly improves the versatility of the scheme.

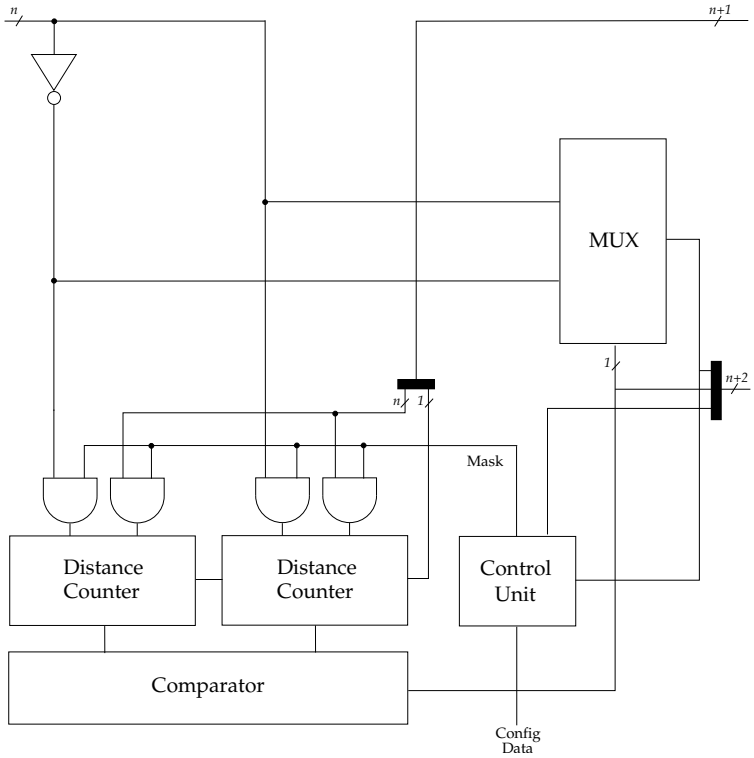


Fig. 5.21: Adaptive PBI (Hamming and Coupling) encoder

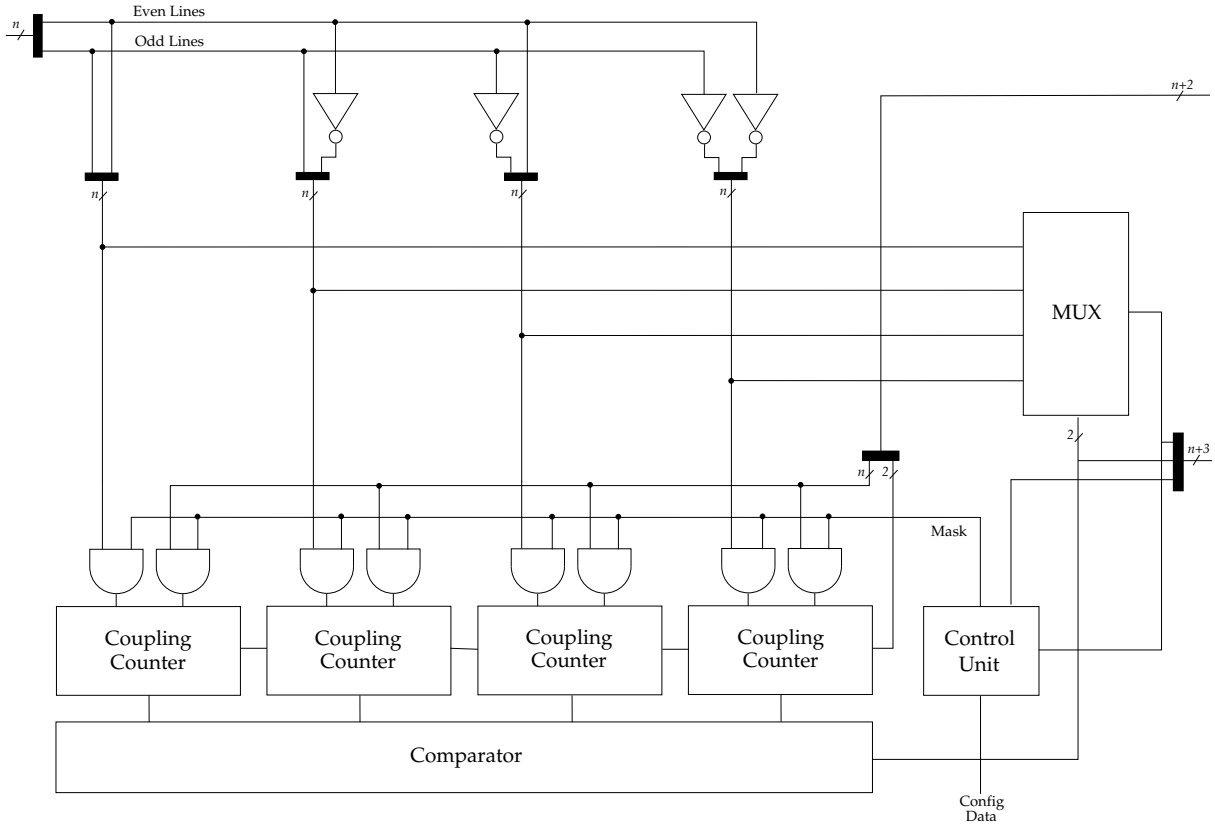


Fig. 5.22: Adaptive POEBI encoder

5.5 Limits for Power Coding

A very important question is related to the theoretical achievable limit for transition activity reduction. In this context, there are two fundamentally different problems to be tackled. On the one hand, finding the minimum achievable self activity can be reduced to a one-dimensional problem, while on the other hand, determining the minimum achievable coupling activity cannot be solved in such a simple way.

5.5.1 Limits for Self Transition Activity

Ramprasad et al. introduced a source-coding framework for the design of self transition activity reducing coding schemes [143]. A particularization of that framework has been used in [15] by Benini et al. in order to develop algorithms for the synthesis of power-reducing encoding and decoding interface logic. In the following, a similar generalized framework is employed. Given the bus width B, the number of possible code words is $N = 2^B$.

The encoder consists of a register bank that stores $m-1$ previous source code words $\{u(i-1), u(i-2), \cdots, u(i-m+1)\}$ used together with $u(i)$ as inputs for the encoding function, an encoding function Ψ_C that generates the encoded word $w(n)$, and a decorrelator whose function is described in the sequel. **Fig**. 5.23 shows the framework particularized for $m = 3$.

Stan and Burleson mentioned in [189] that in contrast to low-power encoding schemes for level signaling which require a complex one-to-many context-dependent correspondence between coded words and source words, transition signaling based schemes can be done in a one-to-one context-independent manner. Transition signaling is straightforward to implement as it consists of an XOR-based decorrelator-correlator structure. The main advantage of using a correlator-decorrelator scheme is that in this way the problem

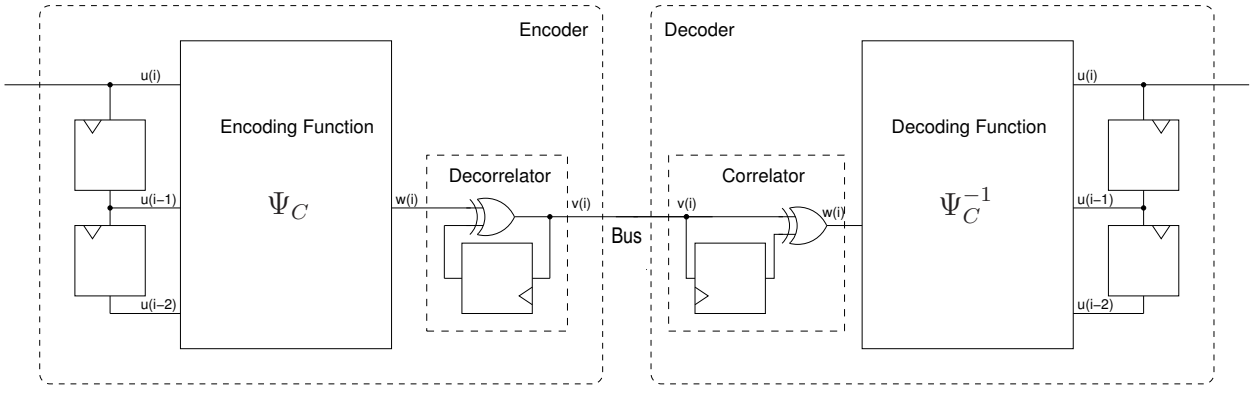


Fig. 5.23: General framework for $m = 3$

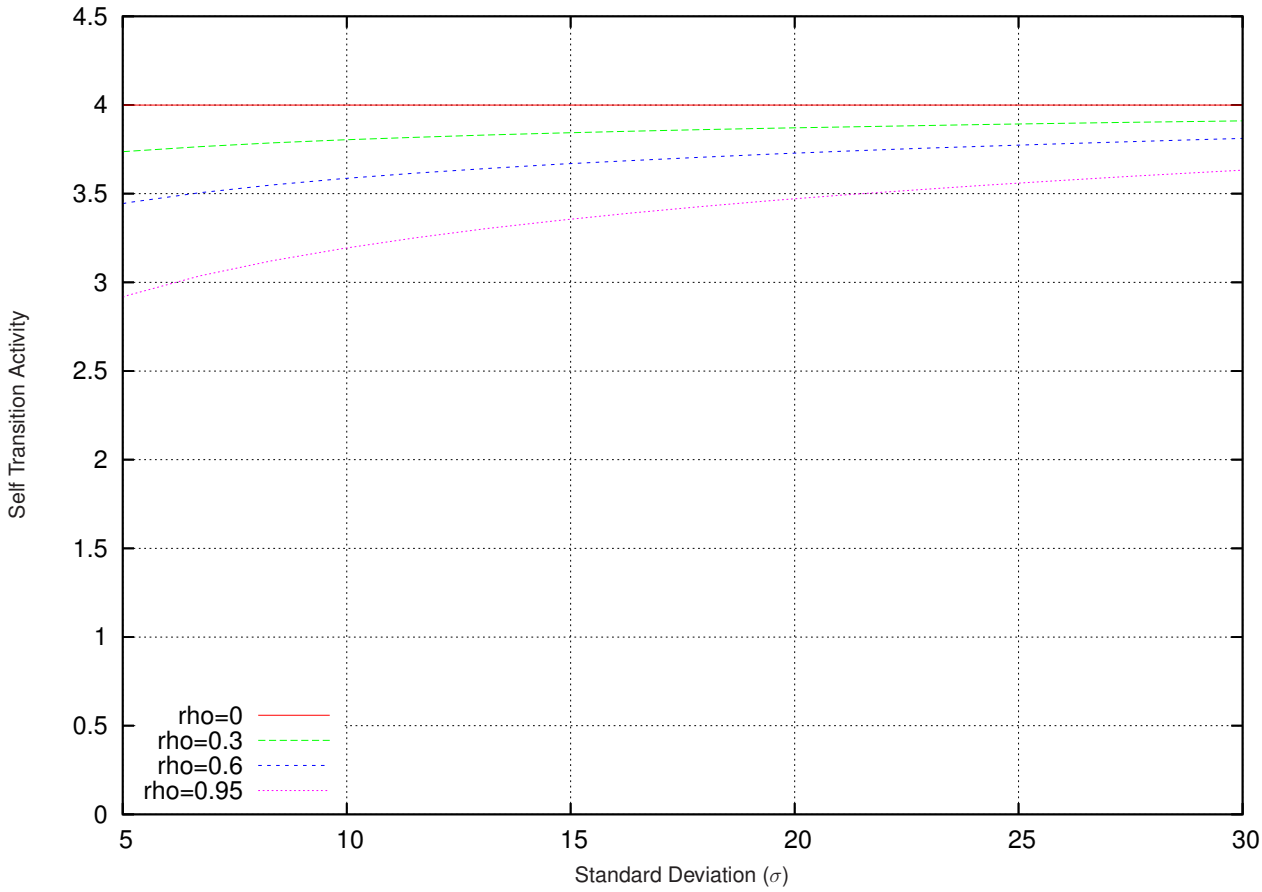


Fig. 5.24: Power cost for Gaussian signal without encoding

of minimizing the transition activity on the bus gets reduced to the problem of minimizing the number of ones on the decorrelator input [15, 189].

The decoder is composed of a correlator that reconstructs $w(i)$ and a decoding function Ψ_C^{-1} whose task is to calculate the initial source code word $u(i)$ as a function of $w(i)$ and $\{u(i-1), u(i-2), \cdots, u(i-m+1)\}$. The efficiency of the encoding function Ψ_C increases with m, as higher m-s allow a better exploitation of data dependencies. Nevertheless, with increasing m, the complexity of the hardware rapidly explodes, which is equivalent to higher area penalties and codec-induced power consumption. In practice, we therefore typically have $m \leq 3$. It is to be mentioned that the final value for m also depends on the data correlation, since in the case of weakly correlated data, a high value of m is not bringing improvements. As illustrated in **Fig**. 5.23, in this work we use encoding frameworks for $m = \{1, 2, 3\}$. Thus, in order to reduce the power consumption induced on the bus by the temporal transition activity, the encoding function Ψ_C has to reduce the average number of ones at its output, i.e. at the input of the decorrelator. Furthermore, the encoding function must guarantee that the decoding function Ψ_C^{-1} uniquely decodes the received value.

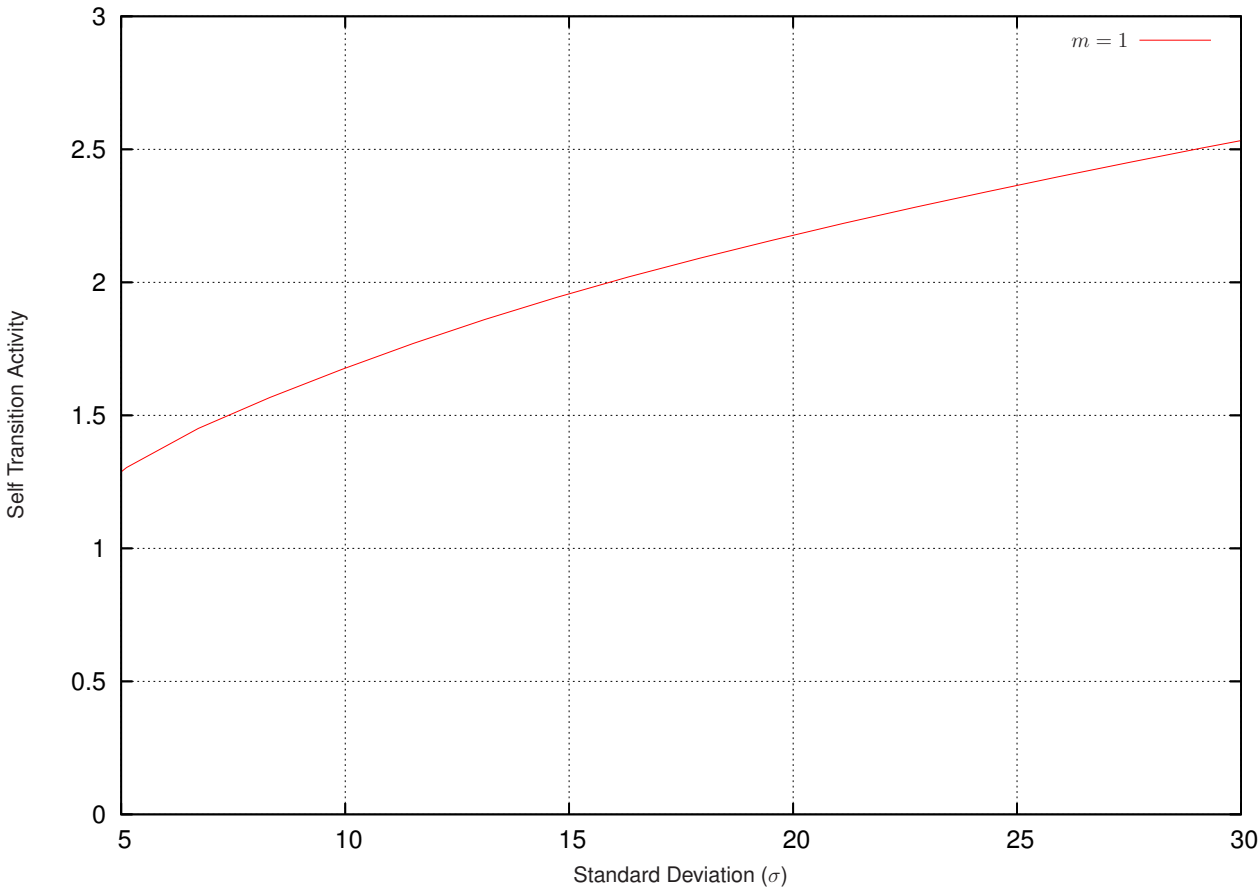


Fig. 5.25: Minimum power cost (inferior limit) for $m = 1$

The exact low transition encoding algorithm for $m = 2$ presented in [15] consists mainly of the table ordering after decreasing probability of appearance of all N^2 possible input pairs and the top-bottom assigning of the code words comprising the lowest possible number of ones, i.e with lowest power cost. Moreover, in order to achieve full decodability, after each assignment the coding table is completely scanned and a so-called conflict array is updated. The algorithm is constructing thus for a given probability distribution, the best possible power-reducing coding scheme, and gives the minimum achievable power cost through coding. Notice that for $m = 2$, complete statistical characterization means the knowledge of the joint probability distribution, i.e. the joint probability matrix.

The algorithm employed in this thesis for computing the minimum power cost is similar to the aforementioned one, though conceptually and computationally simpler and more efficient. For $m = 1$, the encoding algorithm reduces to the ordering in a vector of the source code words after decreasing probability and the sorting in another vector of the encoded words after increasing energy costs (number of comprised ones). The total power cost of the encoding scheme is then given by the scalar multiplication of the two sorted vectors.

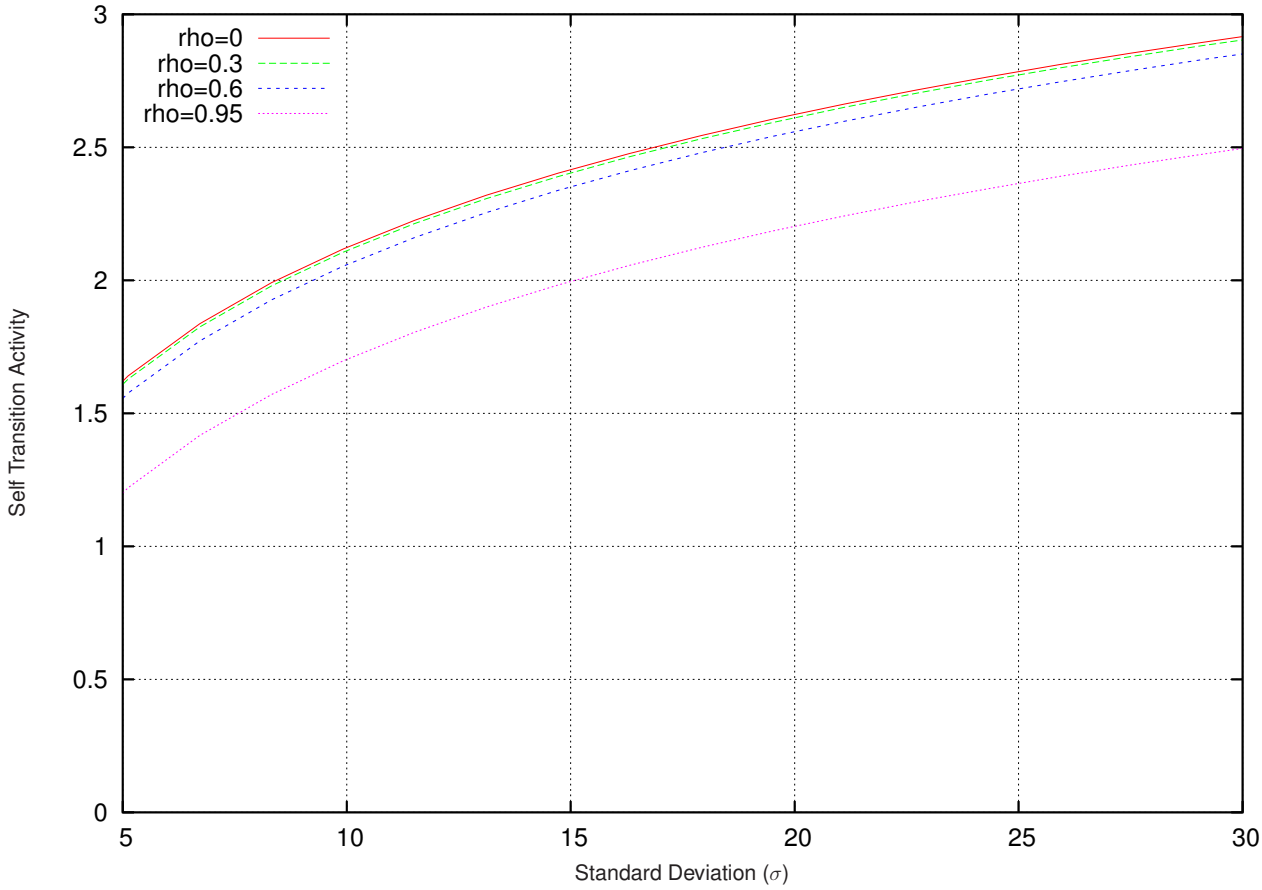


Fig. 5.26: Power cost for the K1 code

For $m = 2$, the algorithm employed in this work does not sort all the N^2 values of the probability matrix $P = [p_{ab}]$, where $p_{ab} = \Pr(u(i-1) = a \ \wedge \ u(i) = b)$ for $a, b = \overline{1, N}$. We only order each row of the probability matrix after decreasing probabilities. By building a vector out of the sum of the column values and scalarly multiplying that vector with the sorted energy cost vector for $\{w(i)\}$ we obtain the minimum power cost. The algorithm is in this case a generalization of the algorithm for $m = 1$. Further, if $m = 2$, then the probability matrix is 3-dimensional and we have to sort all N^2 rows of the probability cube, and add all slices for obtaining the probability vector of $\{w(i)\}$. We expect that the efficiency of encoding schemes improves due to the fact that the statistical characteristics of the signal can be better exploited. Nevertheless, the achieved efficiency comes at the expense of significant complexity penalties.

Additionally, even though our goal was not to construct the power optimal code but only to find the limit when using that best code, the developed algorithm can also be used to generate the optimum encoding function Ψ_C as in [15]. Nevertheless, the proposed algorithm is conceptually slimmer and more efficient. First, the sorting of N^m m-tuples is replaced by N^{m-1} sorting operations of N elements. Second, no table scanning procedure is required as no conflicts can appear.

In the following, several experimental results regarding power cost limits for encoding schemes with $m = 1$ and $m = 2$ are discussed. Additionally, a simple encoding scheme with respect to the lower power cost bounds and the power costs without encoding is analyzed. As input signals zero-mean Gaussian distributed input signals with standard deviation σ varying from 5 to 30 and correlation factor $\rho = \{0, 0.3, 0.6, 0.95\}$ are considered while the bus width is set to $B = 8$.

Fig. 5.24 shows the power cost without employing any encoding scheme, i.e. for $m = 1$. Due to the monotonic character of the power cost function, the following two general observations can be done, which hold for all subsequent cases. First, when the correlation increases, the power consumption decreases due to the fact that fewer bits toggle, and thus the transition activity is smaller. Second, for higher standard deviations more ones in the encoded words appear, increasing the minimum achievable power cost. The minimum power cost for $m = 1$ is represented in **Fig**. 5.25. Even though in this case no encoding scheme can make use of previous input knowledge, it is still possible for the optimal encoding scheme to significantly reduce the power cost.

In contrast, when $m = 2$, the encoding schemes can make use of one previous code word. Consequently, we expect a significant improvement in the minimum achievable power cost. As seen in **Fig**. 5.27, for highly correlated signals, the encoding-based minimum achievable power cost improves. It is important to notice, that when the input signal is uncorrelated, there is no statistical information for any encoding scheme with $m = 2$ to exploit. Therefore, the power cost limit is identical to the case when $m = 1$. This is an expected result, as it is just stating that the achievable power cost limit through encoding schemes is a direct function of the signal correlation. If $u(i)$ and $u(i-m)$ are not correlated, the use of $m+1$ inputs for the encoding function does not improve the power cost limit.

The K1 code has been developed to reduce the transition activity in the MSBs especially for small standard deviations. **Fig**. 5.26 shows that in comparison with the uncoded case, the power cost is decreased. The code behaves very well for poorly correlated signals. However, when the data is highly correlated, even though the switching activity is reduced, the gap to the power cost optimal code is increasing.

To summarize, the power cost limits in the case of zero-mean Gaussian signals depend on the standard deviation and the correlation factor. When the standard deviation increases, the power cost limits also increase, and for increasing correlation factors, the power cost limits are decreasing.

5.5.2 Limits for Total Transition Activity

It is to be noticed that when taking into consideration also the coupling capacitances, the coupling transition activity becomes comparable to the temporal transition activity induced by the capacitances between lines and ground [228]. In this case, the above-

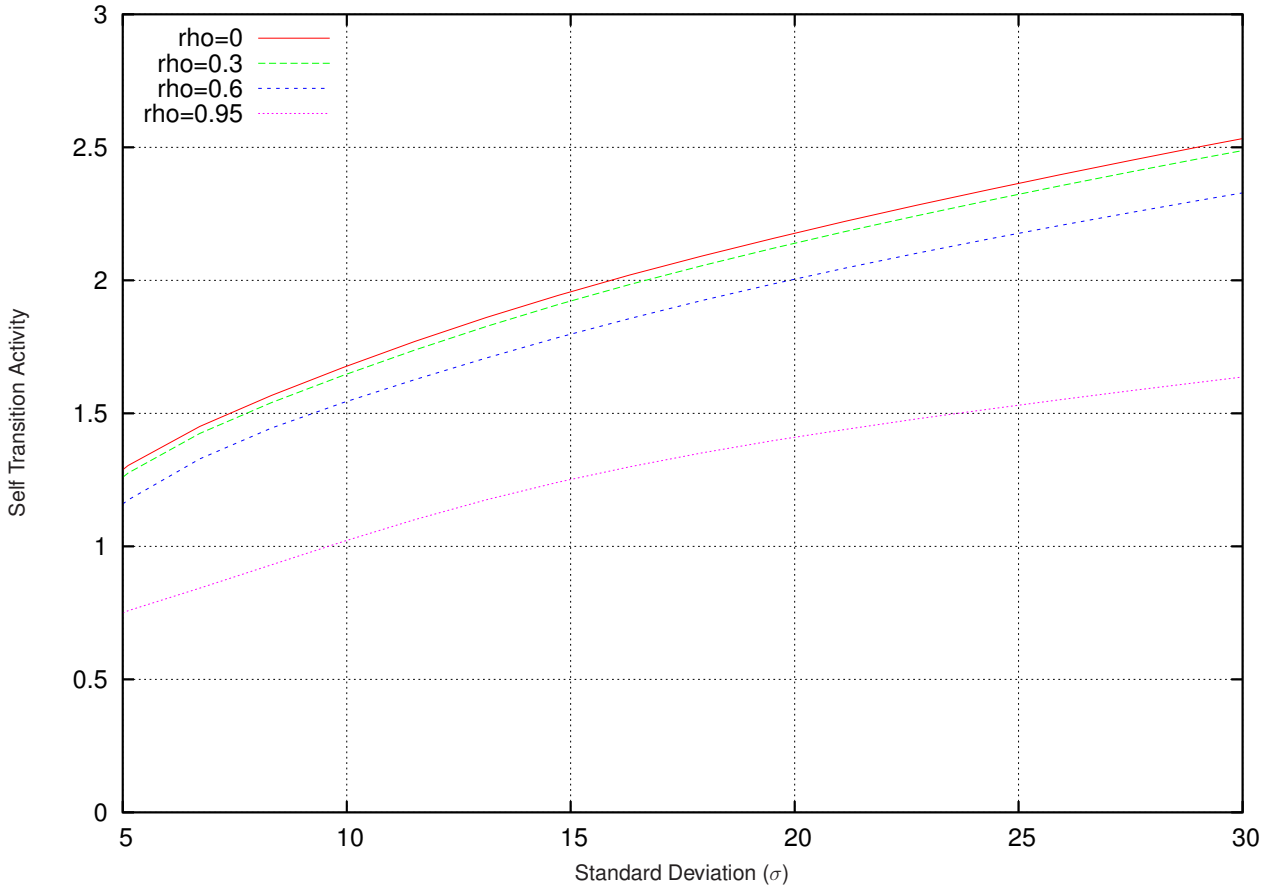


Fig. 5.27: Minimum power cost (inferior limit) for $m = 2$

mentioned decorrelator-correlator structure cannot be employed in order to reduce the complexity of the transition activity minimization problem.

In order to provide a mathematical foundation for the study of different coding strategies, Ramprasad et al. introduced an entropy-based limit for the self activity. However, that method is not suitable for very deep sub-micron buses characterized by high aspect ratios.

Let X and X_{opt} be an ergodic discrete random variable with an entropy rate H_r and the lag-one discrete and stationary Markov process of maximum entropy rate for a given power cost C. Let Y be a lag-one Markov process with the same probability and conditional probability as X. The power costs of X and Y are identical, however, since conditioning reduces entropy [121, 136]:

$$H(X_n|X_{n-1}, \ldots, X_1) \leq H(X_n|X_{n-1}) = H(Y_n|Y_{n-1}). \tag{5.22}$$

Thus, the entropy rate of X is smaller than that of Y. Moreover, since Y is a lag-one discrete Markov process, its entropy rate cannot be larger than that of X_{opt}.

Let $H_C(c)$ be the maximum entropy of an ergodic lag-one Markov process with power cost c. It can be shown, that for each ergodic random process with entropy rate H, it is

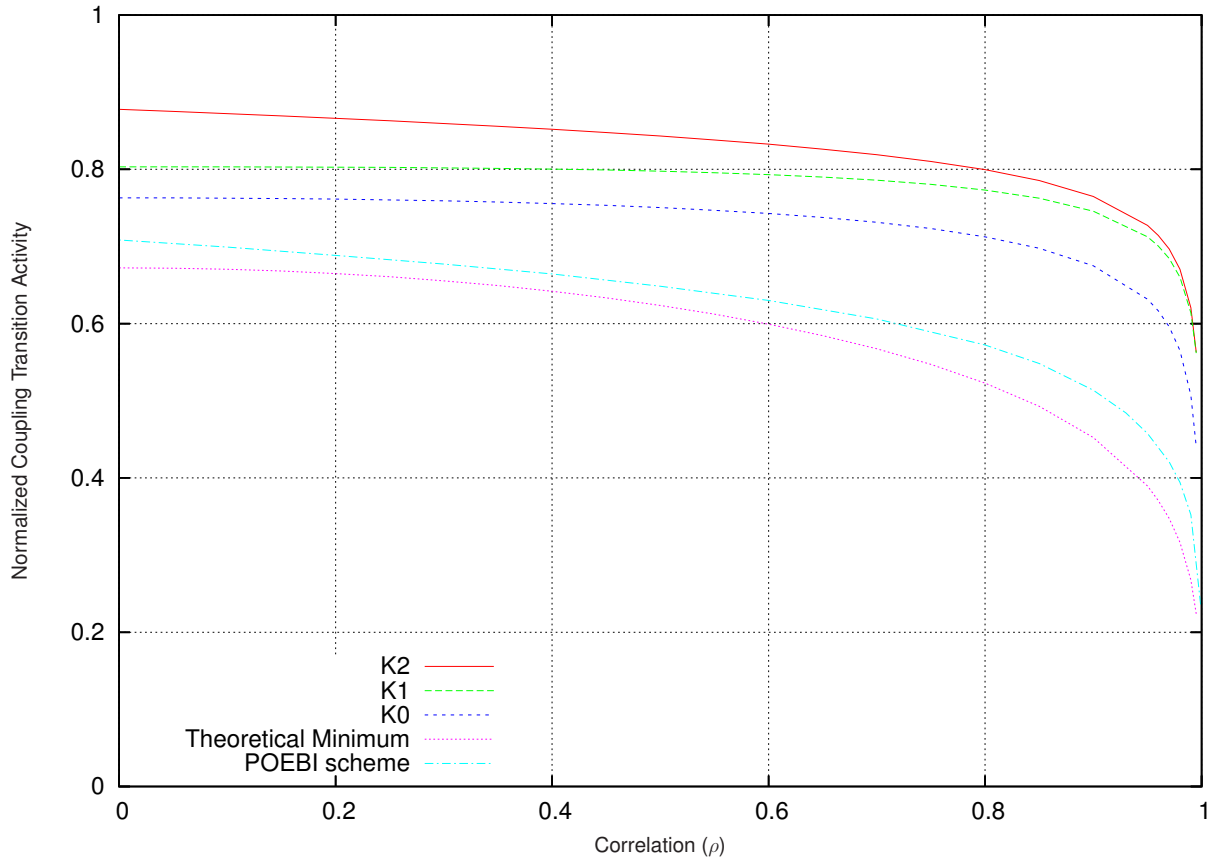


Fig. 5.28: POEBI and minimum power cost

possible to find a coding which achieves asymptotically a cost c that satisfies $H_C(c)$. On the one hand, the Asymptotic Equipartition Property [136] says that for a large n, there is a typical set of 2^{nH} different symbols to transmit. On the other hand, there can be drawn n samples from the Markov process X_{opt} that has entropy rate $H_C(c)$. Further, the Asymptotic Equipartition Property says that there can be obtained $2^{nH(c)}$ different codewords [175]. Thus, a one-to-one code is obtained when $H_C(c) = H$ and the transmission over the bus of the constructed codewords has the power cost c.

Fig. 5.28 illustrates the minimum achievable coupling power cost for varying ρ in the case of an n-bit zero-mean Gaussian signal with $\sigma_n = 0.3125$. The signal has been coded with the non-redundant schemes K0 and K1 as well as with K0POEBI3. It can be noticed that the POEBI scheme – even though a scheme of relatively low complexity – is very effective, as it achieves a total coupling power remarkably cost close to the minimum achievable one. Thus, it can be concluded that in spite of their rather low complexity, the coding schemes developed in this chapter for low-power DSP systems are very powerful and versatile.

5.6 Summary

Signal encoding can be employed for efficiently reducing the self and coupling transition activities, and thus the switching component of the dynamic power consumption. In order to construct efficient codes, the bit-level and word-level activities must be estimated and analyzed.

The first contribution of this chapter is the conclusion drawn after analyzing the bit-level activities in typical DSP signals that bus invert schemes can be efficiently employed in order to decrease the transition activity in the least significant bits. Secondly, it has been shown that the bus invert schemes do not employ or preserve the high correlation typical for the most significant bits. Consequently, by combining non-redundant codes (K0, K1, K3) with bus invert schemes, several highly efficient codes have been developed (PBIH, PBIC).

Another important contribution is the construction of the partial OEBI (POEBI) encoding scheme. In this context, significant improvements, especially concerning the coupling transition activity, have been reported. Additionally, two simple yet very effective schemes based on bus invert (APBI) and OEBI (APOEBI) are developed. The main idea of those adaptive codes is to estimate the breakpoints by monitoring only a selected set of bus lines and not all as in previously described codes.

Finally, fundamental limits for both self and total activity have been derived and analyzed proving thus the effectiveness of the developed codes.

Chapter 6

Signal Encoding for Performance Optimization

Contents

Sotiriadis and Chandrakasan proposed in [179] to employ coding for increasing the data throughput in deep sub-micron buses. The fundamental idea is to avoid those transition patterns that cause the largest delays on the bus. On the one hand, by eliminating the worst case patterns, the clock period can be reduced, thus increasing the throughput. On the other hand, because of the imposed constraints in terms of transition patterns, the maximum achievable data rate in a clock period is reduced in the case of a fixed-width

bus. Therefore, coding for improving throughput is applicable only if the clock period can be decreased at a higher rate than the drop in maximum achievable data rate. In the rest of this work, *coding for speed*, *coding for performance*, and *coding for throughput* are interchangeably used.

Contrary to the low-power coding problem, treating coding for throughput in an application-specific manner does not generally bring about any advantage. In order to avoid a worst case delay, it is mostly of no interest to know or estimate the frequency of appearance of the delay patterns responsible for those worst cases. Coding for throughput can be mainly implemented in two ways: by avoiding all worst case delay transition patterns or by prohibiting a set of states involved in those worst case transitions. As shown later throughout the chapter, there are various benefits as well as downsides related to the two abovementioned methods.

This chapter is organized as follows. First, **Sec**. 6.1 analyzes and classifies the switching patterns in terms of delay in capacitively and inductively coupled buses and discusses pros and cons related to coding schemes based on avoiding transitions and states. On this basis, **Sec**. 6.2 deals with limits related to transition-based and state-based coding for performance improvement. Bounds and closed form solutions thereof are derived both from recurrent transition and state counting as well as from the theory of noiseless constrained channels. In addition, simple coding schemes are proposed and analyzed in **Sec**. 6.3. Moreover, coding for speed is compared and combined with classic anti-crosstalk techniques like spacing and shielding. **Sec**. 6.4 shows the relation between coding for power and coding for performance. In this context, it is shown how both types of coding can be combined.

6.1 Improving Throughput in Buses by Coding

Coding for throughput improvement and coding for crosstalk are two interrelated problems. As in the case of coding for performance, the fundamental idea of coding for crosstalk is to avoid those transitions and/or states responsible for the largest crosstalk-induced noise. For example, in the case of simultaneous switching in capacitively coupled buses, the largest crosstalk-induced noise and the largest delay appear in lines which oppositely toggling neighbors. The two problems are thus equivalent and due to their similarity, the focus is set on coding for performance.

In the following, we analyze the pattern dependency of the delay classes and show how coding for speed can be implemented by prohibiting transitions and states that induce worst case delays.

6.1.1 Delay Classes

Let us consider for simplicity of formalism that $\tau_0 = 1$ and $\Delta b_i = 1$, where line i is the line under analysis. As shown in **Chap**. 4, the delay in capacitively coupled buses can be expressed as:

$$\delta_i = (1 + 2\kappa) - \kappa(\Delta b_{i-1} + \Delta b_{i+1}), \tag{6.1}$$

where δ_i is the delay in line i. Thus, there are five possible delay classes for a switching line from 1 to $1 + 4\kappa$ in steps of κ. For $k = \overline{1,5}$, we can define the delay class Δ_k as a set

$$\Delta_k = \{\underline{\Delta b} \mid \max \delta_i = 1 + (k-1)\kappa, \text{ for all } i = \overline{1,B}\} \tag{6.2}$$

where B is the bus width. The delay $1 + (k-1)\kappa$ is called the characteristic delay of the delay class Δ_k. In the sequel, we denote with δ_M the maximum delay in a bus for a transition pattern, that is M is the index of the line with the largest delay. If considering also the zero delay class, Δ_0, we can define the six delay classes as follows:

$$\begin{aligned}
\Delta_5 &: \ \Delta b_M \neq 0 \text{ and } \Delta b_{M-1} + \Delta b_{M+1} = -2\Delta b_M \\
\Delta_4 &: \ \Delta b_M \neq 0 \text{ and } \Delta b_{M-1} + \Delta b_{M+1} = -\Delta b_M \\
\Delta_3 &: \ \Delta b_M \neq 0 \text{ and } \Delta b_{M-1} + \Delta b_{M+1} = 0 \\
\Delta_2 &: \ \Delta b_M \neq 0 \text{ and } \Delta b_{M-1} + \Delta b_{M+1} = \Delta b_M \\
\Delta_1 &: \ \Delta b_M \neq 0 \text{ and } \Delta b_{M-1} + \Delta b_{M+1} = 2\Delta b_M \\
\Delta_0 &: \ \Delta b_M = 0
\end{aligned} \tag{6.3}$$

Sotiriadis pointed out in [175] that when crosstalk is important, the delay class Δ_0 is not correctly defined. Even though a line does not switch, the coupling with the neighbors can be so high that a toggle in an aggressor generates a temporary swing on the victim that cannot be neglected. Therefore, we can define instead of Δ_0 three so-called zero delay classes as follows:

$$\begin{aligned}
\Delta_{00} &: \ \Delta b_M = 0 \text{ and } \Delta b_{M-1} = \Delta b_{M+1} = 0 \\
\Delta_{01} &: \ \Delta b_M = 0 \text{ and } |\Delta b_{M-1}| + |\Delta b_{M+1}| = 1 \\
\Delta_{02} &: \ \Delta b_M = 0 \text{ and } |\Delta b_{M-1}| + |\Delta b_{M+1}| = 2
\end{aligned} \tag{6.4}$$

The delay of class Δ_{0k} is thus $k\kappa$, for $k = \{0, 1, 2\}$. Actually, the zero delay classes must be taken into consideration separately only in encoding schemes which try to avoid delays less than or equal to $1 + \kappa$. As indicated in the next sections, these encoding schemes generally reduce the data rate capacity too much to be of high interest.

As also seen in **Chap**. 4, the aforementioned partitioning of delay classes does not hold when inductive effects cannot be neglected anymore. When only the inter-wire capacitances have to be taken into consideration, all possible delay values are very tightly closed around the characteristic values of the delay classes. However, with increasing inductive effects, the delays start to detach from their characteristic values and the delay classes begin to dissolute. Thus, delay classes are rather intervals than fixed values.

Tab. 6.1: Comparison between delay classes/intervals for capacitive and inductive coupling

Capacitive	*Capacitive and Inductive*
$1+4\kappa$	$1+4\kappa-2\alpha_{k,L}^{(1)}\pm\eta$
$1+3\kappa$	$1+3\kappa-\alpha_{k,L}^{(1)}\pm\eta$
$1+2\kappa$	$1+2\kappa\pm\eta$
$1+\kappa$	$1+\kappa+\alpha_{k,L}^{(1)}\pm\eta$
1	$1+2\alpha_{k,L}^{(1)}\pm\eta$

In the sequel, the effect of increasing inductive coupling on delay classes is analyzed by revisiting the ELD model introduced in **Chap**. 4. Conceptually, the coefficients of the ELD model can be split in two parts: on the one hand, the coefficients standing for capacitive coupling ($\alpha_{ij,C}$), and on the other one, the coefficients for the inductive coupling ($\alpha_{ij,L}$). Thus:

$$\alpha_{ij}=\alpha_{ij,C}+\alpha_{ij,L},\ i\neq j, \tag{6.5}$$

where $\alpha_{ij,C}\leq 0$ and $\alpha_{ij,L}\geq 0$. Capacitive coupling is a short-range effect and thus, only first-order neighbors can be considered as in the models proposed in [175, 181]. Consequently, we can write the delay in line k as:

$$\begin{aligned}\delta_k=\alpha_k\Delta b_k^2&+(\alpha_{kk-1,C}\Delta b_{k-1}+\alpha_{kk+1,C}\Delta b_{k+1})\Delta b_k\\&+(\alpha_{kk-1,L}\Delta b_{k-1}+\alpha_{kk+1,L}\Delta b_{k+1})\Delta b_k\\&+\sum_{i\neq 0,1}(\alpha_{kk-i,L}\Delta b_{k-i}+\alpha_{kk+i,L}\Delta b_{k+i})\Delta b_k\end{aligned} \tag{6.6}$$

In a symmetric bus, α_{kk-i}=$\alpha_{kk+i}\overset{\text{def}}{=}\alpha_k^{(i)}$, if the corresponding neighbor exists. We can define in a similar way $\alpha_{k,C}^{(i)}$ and $\alpha_{k,L}^{(i)}$.When the corresponding neighbors do not exist, either the coefficients or the associated transitions can be defined as zero. For a symmetric bus, the delay becomes:

$$\begin{aligned}\delta_k &= \alpha_k\Delta b_k^2+(\alpha_{k,C}^{(1)}+\alpha_{k,L}^{(1)})(\Delta b_{k-1}+\Delta b_{k+1})\Delta b_k\\&\quad+S_{ind}(k)\Delta b_k,\end{aligned} \tag{6.7}$$

where $S_{ind}(k)$ stands for the cumulative influence of the inductive aggressors of an order higher than two:

$$S_{ind}(k)=\sum_{i\neq 0,1}(\alpha_{kk-i,L}\Delta b_{k-i}+\alpha_{kk+i,L}\Delta b_{k+i}). \tag{6.8}$$

Let $\eta\overset{def}{=}\max\{S_{ind}(k)\}=\max\{\sum_{i\geq 2}(\alpha_{kk-i,L}+\alpha_{kk+i,L})\}\geq 0$ be the maximum cumulative effect of the inductive aggressors. For a symmetric bus, $\eta=2\max\{\sum_{i\geq 2}\alpha_{k,L}^{(i)}\}$. It can be noticed that the non-zero delay classes are modified under inductive coupling from fixed values to intervals as shown in **Tab**. 6.1. Several major cases with regard to the relationship between inductive and capacitive coupling can be identified:

- $\kappa \gg \alpha_{k,L}^{(1)} + 2\eta$: in this case, the capacitive coupling completely dominates the inductive one which can be neglected without any accuracy loss.
- $\kappa \gtrsim \alpha_{k,L}^{(1)} + 2\eta$: the inductive coupling cannot be neglected; this case encompasses low-medium inductive coupling scenarios and as shown later, it corresponds to the case of disjoint delay classes.
- $\kappa \lesssim \alpha_{k,L}^{(1)} + 2\eta$: both inductive and capacitive couplings cannot be neglected, the inductive coupling is getting more important, and the delay classes are not disjoint anymore; this case corresponds to higher inductive coupling and is less probable in reality.
- $\kappa \ll \alpha_{k,L}^{(1)} + 2\eta$: the inductive coupling outweighs the capacitive one and the delay classes are totally mixed; this case is very unrealistic in normal on-chip buses unless systems are designed for this purpose.

Generally speaking, the least inductive cases appear when the capacitive influence of the first-order neighbors outweighs the cumulated effect of all inductive aggressors. As also mentioned in **Chap**. 4, the first two cases are the most realistic ones in on-chip interconnects, since the other two situations are mostly avoided in practice because of the high induced crosstalk [25, 192].

6.1.2 Transition and State Coding

In a classically operating bus, the clock period, T_{ck}, must be chosen to be large enough so that any transition can be completed, i.e.

$$T_{ck} \geq \tau_0(1 + 4\kappa). \tag{6.9}$$

By prohibiting transitions from higher delay classes, the lower bound required for T_{ck} gets reduced and the bus can be clocked faster. For instance, by avoiding class Δ_5, the bus can be clocked $\frac{1+4\kappa}{1+3\kappa}$ times faster. For $\kappa \gg 1$, this means approximately 1.33 times faster. If we also interdict Δ_4, the speed can be increased about 2 times. Moreover, by allowing only Δ_0, Δ_1, and Δ_2, the bus can be theoretically sped up 4 times.

Basically, there are two ways in avoiding delays of certain classes. One the one hand, one can try to avoid directly those transitions, and on the other hand, a selected set of states involved in those toggling patterns can be eliminated from the symbols alphabet. The main advantage of the state-based coding is its intrinsic simplicity. The codec is at most a static map. In a dynamic environment, for instance in applications characterized by different entropy rates and running on a reconfigurable platform, the bus width is generally predetermined and fixed. Such static maps can be defined for each application and loaded whenever the associated application is running. On the other hand, the versatility of transition-based coding allows the construction of more efficient encoding schemes. As expected, the benefit of flexibility is paid in terms of implementation complexity.

Tab. 6.2: Transition and state avoiding in a 4-bit wide bus

	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0000																
0001																
0010																
0011																
0100																
0101																
0110																
0111																
1000																
1001																
1010																
1011																
1100																
1101																
1110																
1111																

The versatility of transition-based coding is due to the fact that impossible transitions can be theoretically mapped on unallowed ones. However, in state-based coding, by not allowing one state, all transitions in which that state is potentially participating are automatically interdicted. Thus, it can happen that more transitions than required are prohibited. It is to be mentioned, that if for certain applications any transition pattern is possible, then transition-based coding is not applicable and the only possible coding is the state-based one.

Consider a similar example as in [175], that is a 4-bit wide bus with no coupling at the margins. We want to avoid the delay classes Δ_5 and Δ_4, which means that for $\kappa = 4.5$, the speed can be increased 1.9 times. If we prohibit the states 0010, 0100, 0101, 1010, 1011, and 1101 (i.e. the rows and columns marked light in **Tab**. 6.2), then the worst case delay is limited to $1 + 2\kappa$. As there are 10 allowed states, the number of bits that can be transmitted each clock cycle is reduced from 4 to $\log_2 10 = 3.32$, that is by a ratio of about 1.2. Consequently, a maximum throughput improvement of around $1.9/1.2 \simeq 1.583$ can be achieved. In order to obtain a trivial static map for the codec scheme, the set of states can be further decreased to 8 and the improvement factor is 1.42. It is to be noticed that a bus with one extra line results in more closely inter-spaced lines if the physical width is limited to a fixed value. Thus, the value of κ increases and the net result in data rate improvement is reduced by a factor depending on the wire geometry.

By interdicting only the Δ_4 and Δ_5 transitions (i.e. the transitions marked dark gray in **Tab**. 6.2), one could map transitions with probability zero on the prohibited ones. Alternatively, intermediate states can be inserted between the states involved in a restricted transition similarly to the so-called stutter coding [91] in order to obtain two or more al-

lowed transitions. Afterwards, low-probable transitions can be mapped on such extended transitions in order to keep the introduced redundancy low.

It is nevertheless obvious, that by not permitting a set of transitions and/or states, the maximum achievable information rate on the bus is reduced. For an n-bit wide bus, the bit rate reduction factor, $\zeta_b(n,k)$, is defined as the ratio between the maximum achievable information rate on the coded bus and the actual bus width. In the next section, exact values and bounds for the bit rate reduction factor for transition and state coding are derived. Further, we can define also the speed increasing factor, $\zeta_s(n,k)$, which stands for the interconnect delay decreasing rate:

$$\zeta_s(n,k) = \frac{1+4\kappa_n}{1+k\kappa_n}, \tag{6.10}$$

where $k = \{0,1,2,3,4\}$ indicates the highest allowed delay. It is to be noticed that for a fixed physical bus width, κ is actually a function of n, therefore the simpler notation κ_n. In the rest of the work, the notations κ and $\zeta_s(k)$ are used whenever the bus width is not subject to modifications.

For an efficient encoding, we have in general $k = \{2,3\}$ [175]. Thus, the total actual throughput increase rate can be defined as:

$$\zeta_t(n,k) = \zeta_s(n,k) \cdot \zeta_b(n,k). \tag{6.11}$$

In order for a code to be efficient, the achieved throughput increase rate must be higher than one, i.e. $\zeta_t(n,k) > 1$.

In the case of non-negligible inductive coupling, the delay classes given in **Tab**. 6.1 are disjoint only if $\alpha_k^{(1)} \leq -2\eta$, for all $k = \overline{1,B}$. In this case, coding for performance is done by applying the same mapping schemes as for buses characterized only by inter-wire capacitance, otherwise the mappings have to be adjusted. These are in the most general case simple static maps and without any loss of generality, we consider only disjoint delay classes.

In the case of inductive coupling, $\zeta_b(n,k)$ is the same as for the capacitive case. However, in the case of disjoint classes we have

$$\zeta_s(n,k) = \frac{1+4\kappa_n - 2\alpha_{k,L}^{(1)} + \eta}{1+k\kappa_n - (k-2)\alpha_{k,L}^{(1)} + \eta}. \tag{6.12}$$

where $\tau_0 = 1$ and $k = \overline{0,3}$. For instance, when $k = \{2,3\}$, the equality becomes:

$$\zeta_s(n,2) = \frac{1+4\kappa_n - 2\alpha_{k,L}^{(1)} + \eta}{1+2\kappa_n + \eta}, \tag{6.13}$$

$$\zeta_s(n,3) = \frac{1+4\kappa_n - 2\alpha_{k,L}^{(1)} + \eta}{1+3\kappa_n - \alpha_{k,L}^{(1)} + \eta}, \tag{6.14}$$

$$\zeta_s(n,4) = 1. \tag{6.15}$$

It can be easily shown that coding for performance would be more efficient with inductive coupling only if $\kappa \leq -\alpha_{k,L}^{(1)}/(2\alpha_{k,L}^{(1)} + \eta) \leq 0$. Nevertheless, κ is a nonnegative parameter, which renders the aforementioned inequality impossible. For $k = \{2, 3\}$ thus, even in the case of a low-medium inductive coupling, the possibilities to increase throughput deteriorate in comparison with the non-inductive case. Consequently, the effectiveness of coding schemes for throughput improvement must be assessed at high levels of abstraction especially in the case of inductive coupling and the required information for this purpose is intrinsically comprised in the ELD model.

6.2 Fundamental Limits of Coding for Throughput

It has been already pointed out, that by not permitting a certain set of transitions, the maximum achievable information rate on the bus is reduced. In the following, we derive the number of allowed states in a recurrent manner and calculate thus the maximum achievable information rate. We also calculate the number of permitted transitions and derive bounds for the bus capacity that are more restrictive than previously developed ones. In order to determine the exact capacity of encoded buses, we employ the theory of noiseless constrained channels. First, we focus on the more simple case of prohibiting states involved in transitions that imply unwanted transitions and secondly, we address the issue of restricting only transitions.

6.2.1 Limits for State Coding

Because of the fact that only a certain set of transitions are allowed, a bus can be regarded together with the coding for performance scheme as a discrete noiseless constrained channel. As shown in [53, 82, 121, 168], once the associated adjacency matrix is constructed, the maximum achievable information rate, i.e. the capacity ϱ of the noiseless constrained channel, is given by the logarithm of the spectrum of the adjacency matrix:

$$\varrho = \log_2 \rho(A_G), \tag{6.16}$$

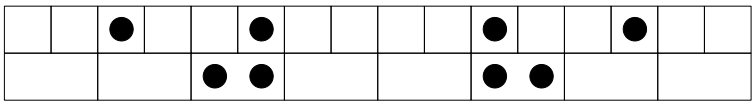
where A_G represents the adjacency matrix and $\rho(A_G)$ denotes the spectrum of A_G, i.e. its maximum absolute eigenvalue (see **App**. C). Further, the bit rate reduction factor for an n-bit wide bus can be defined similarly as the bus utilization factor introduced in [175]:

$$\zeta_b(n, k) = \frac{\log_2 \rho(A_{Gnk})}{n}. \tag{6.17}$$

where A_{Gnk} denotes the adjacency matrix for an n-bit wide bus as a function of the maximum permitted delay class. In the following, we use in the case of fixed values of k the simpler notation A_{Gn} for the adjacency matrix.

Tab. 6.2 shows the states that have to be prohibited in the symbol alphabet in order to avoid Δ_4 and Δ_5 transitions in the case of a 4-bit wide bus. Actually, it represents exactly

Fig. 6.1: Recurrent elimination of states for avoiding delay classes Δ_5 and Δ_4



the adjacency matrix of the associated discrete noiseless constrained channel. Moreover, for $n = 3$, we have:

$$A_{G3} = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \end{pmatrix}. \tag{6.18}$$

The spectra of the adjacency matrices are $\rho(A_{G3}) = 6$ and $\rho(A_{G4}) = 10$. For high values of n, it is virtually impossible to calculate the spectrum. However, in the case of state coding we do not have to construct A_{Gn} in order to compute $\rho(A_{Gn})$. We notice that the spectrum is equal to the allowed number of states. This is actually a trivial result both from matrix theory and information theory. Because the same amount of rows and lines have only zeros and all the others only ones, matrix theory tells us immediately that the maximum absolute eigenvalue is equal to the number of non-zero rows (and columns). Furthermore, from an information theory perspective, we can regard the constrained bus as an unconstrained bus with a reduced alphabet. The logarithm of the cardinality of the resulting alphabet is equal to the required number of bits to represent the information, and thus the maximum bit rate achievable on the bus.

Consequently, in order to determine the exact capacity of the bus, the adjacency matrix is not required but the exact number of permitted states, $p_s(n)$[1]. In order to find $p_s(n)$, we have to analyze the way states are prohibited for different values of n. **Fig**. 6.1 shows the recurrent manner states are eliminated in order to avoid delay classes Δ_5 and Δ_4, i.e. $k = 2$ since the maximum allowed delay class is Δ_3. At step n, we split the states in eight categories corresponding to the last three bits of the state index. We have to block the third and sixth category corresponding to 010 and 101, respectively. In order to find out $p_s(n)$, one has to add the number of allowed states at the previous steps in the same categories. Thus, the recurrence must be related to the four categories and we can write:

$$\underline{\psi}_{n+1} = G\underline{\psi}_n, \tag{6.19}$$

where $\underline{\psi}_n$ represents the 1×4 vector indicating the allowed states in each category at step

[1]The number of states, $p_s(n, k)$, is also a function of k; for simplicity however, the notation $p_s(n)$ can be used whenever k has a fixed value

n and

$$G = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}. \tag{6.20}$$

is the recurrence matrix or the generating matrix, G, as seen in **Fig**. 6.1.

Let $\underline{u}_4 = [1111]^t$ be the 1×4 unity vector. Consequently, the total number of permitted states is:

$$p_s(n) = \underline{u}_4^t \underline{\psi}_n = \underline{u}_4^t G^{n-2} \underline{\psi}_2 = \underline{u}_4^t G^{n-2} \underline{u}_4, \tag{6.21}$$

because $\underline{\psi}_2 = \underline{u}_4$.

The generating matrix G is a function of n, k, and the considered margins, and represents a so-called $0-1$ matrix. Thus, G is positive definite and all positive definite matrices are normal. The Spectral Theorem [5] says that for any normal matrix G, there exists a unitary matrix V such that:

$$G = VDV^*, \tag{6.22}$$

where D is the diagonal matrix whose entries are the eigenvalues of G, and V is the matrix with the corresponding eigenvectors as columns. In the case of a unitary matrix we have $V^* = V^{-1}$, where V^* represents the conjugate transpose (also called the Hermitian adjoint) of V [5]. Let λ_1, λ_2, λ_3, and λ_4 be the eigenvalues of G and let $\underline{v}_1$, $\underline{v}_2$, $\underline{v}_3$, and $\underline{v}_4$ be the corresponding eigenvectors. Thus,

$$\begin{aligned} p_s(n) &= \underline{u}_4^t G^{n-2} \underline{u}_4 = \underline{u}_4^t V D^{n-2} V^* \underline{u}_4 \\ &= \sum_{i=1}^{4} v_i[i]^2 \lambda_i^{n-2}, \end{aligned} \tag{6.23}$$

where $v_i[i]$ denotes the i-th element of the i-th eigenvector. In our case, $\lambda_{1,2} = \frac{1 \pm \sqrt{5}}{2}$ and $\lambda_{3,4} = \frac{1 \pm i\sqrt{3}}{2}$. It can be shown that:

$$\begin{aligned} p_s(n) &= v_1[1]^2 \lambda_1^{n-2} + v_2[2]^2 \lambda_2^{n-2} \\ &= v_1[1]^2 (1-\varphi)^{n-2} + v_2[2]^2 \varphi^{n-2} \\ &= \frac{2}{\sqrt{5}} \left[\varphi^{n+1} + (1-\varphi)^{n+1} \right] = 2F_{n+1}, \end{aligned} \tag{6.24}$$

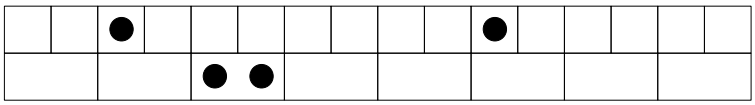
where $\varphi = \frac{1+\sqrt{5}}{2}$ is the so-called golden ratio, and F_n represents the n-th Fibonacci number ($F_{n+1} = F_n + F_{n-1}$ with $F_0 = 0$ and $F_1 = 1$). Consequently, the bit rate reduction factor for state coding that eliminate Δ_5 and Δ_4 classes becomes:

$$\zeta_b(n, 2) = \frac{\log_2 2F_{n+1}}{n} = \frac{1 + \log_2 F_{n+1}}{n}, \tag{6.25}$$

and therefore, we have for very large buses:

$$\lim_{n \to \infty} \zeta_b(n, 2) = \lim_{n \to \infty} \frac{1 + \log_2 F_{n+1}}{n} = \log_2 \varphi \simeq 0.69424. \tag{6.26}$$

Fig. 6.2: Recurrent elimination of states for avoiding delay class Δ_5



In order to have an efficient coding for performance $\zeta_t(n,2)$ must be greater than 1. For large buses, this means that the following condition has to be fulfilled:

$$\kappa > \frac{1-\log_2\varphi}{4\log_2\varphi-2} \simeq 0.39353. \tag{6.27}$$

In large buses with inter-wire capacitance, the best state coding is efficient if κ is higher than the limit calculated above. With increasing inductive effects, that lower bound increases.

Fig. 6.2 shows the recurrent way states have to be interdicted in order to avoid Δ_5 transitions. In this case, $\underline{\psi}_2 = \underline{u}_4$ and

$$G = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}. \tag{6.28}$$

The eigenvalues of G are (see **App**. A):

$$\lambda_1 = 0, \tag{6.29}$$

$$\lambda_2 = \frac{1}{3}\Big(2+\frac{1+i\sqrt{3}\cos 2\phi}{\sin 2\phi}\Big), \tag{6.30}$$

$$\lambda_3 = \frac{1}{3}\Big(2+\frac{1-i\sqrt{3}\cos 2\phi}{\sin 2\phi}\Big), \tag{6.31}$$

$$\lambda_4 = \frac{2}{3}\Big(1-\frac{1}{\sin 2\phi}\Big) = \lambda_3^*, \tag{6.32}$$

where

$$\theta = -\operatorname{asin}\frac{2}{25} \quad \text{and} \tag{6.33}$$

$$\phi = \operatorname{atan}\sqrt[3]{\tan\frac{\theta}{2}}. \tag{6.34}$$

It is to be noticed that λ_2 is a real number ($\lambda_2 \simeq 1.75488$) while λ_3 and λ_4 are complex ones ($\lambda_{3,4} \simeq 0.12256 \pm 0.74486i$). Now, the number of states can be computed as:

$$p_s(n) = \lambda_2^{n-2}(\underline{u}_4^t\underline{v}_2)^2 + 2\mathrm{Re}\{\lambda_3^{n-2}(\underline{u}_4^t\underline{v}_3)^2\}, \tag{6.35}$$

where $\underline{v}_i$ is the eigenvector associated to the eigenvalue λ_i. The spectrum of G is $\rho(G) = \lambda_2$ and thus, for very large buses the bit rate reduction factor:

$$\lim_{n\to\infty}\zeta_b(n,3) = \log_2\rho(G) = \log_2\lambda_2 \simeq 0.81137. \tag{6.36}$$

Therefore, in buses with inter-wire capacitance, the state coding is efficient for large buses only if

$$\kappa > \frac{1 - \log_2 \rho(G)}{4 \log_2 \rho(G) - 3} \simeq 0.76841. \tag{6.37}$$

This state coding scheme would be more efficient than the previous one only if:

$$\kappa \leq \frac{\log_2 \rho(G) - \log_2 \varphi}{3 \log_2 \varphi - 2 \log_2 \rho(G)} \simeq 0.25464. \tag{6.38}$$

Nonetheless, both schemes are ineffective for such values of κ, and we can conclude that in the case of state coding it is more efficient to prohibit states involved in Δ_5 and Δ_4 transitions than only those responsible for the Δ_5 delay class.

When considering inductive coupling, the analysis becomes slightly more complex. The second coding scheme is more efficient than the first one if:

$$\frac{1 + 4\kappa - 2\alpha_{k,L}^{(1)} + \eta}{1 + 3\kappa - \alpha_{k,L}^{(1)} + \eta} \cdot \log_2 \rho(G) \geq \frac{1 + 4\kappa - 2\alpha_{k,L}^{(1)} + \eta}{1 + 2\kappa + \eta} \cdot \log_2 \varphi, \tag{6.39}$$

which is equivalent to:

$$\kappa \leq \frac{[\log_2 \rho(G) - \log_2 \varphi](1 + \eta) + \alpha_{k,L}^{(1)} \log_2 \varphi}{3 \log_2 \varphi - 2 \log_2 \rho(G)} < 0.26(1 + \eta) + 1.51\alpha_{k,L}^{(1)}. \tag{6.40}$$

Further, in order for the coding to be effective, the total throughput increase rate must be greater than one:

$$\frac{1 + 4\kappa - 2\alpha_{k,L}^{(1)} + \eta}{1 + 3\kappa - \alpha_{k,L}^{(1)} + \eta} \cdot \log_2 \rho(G) \geq 1, \tag{6.41}$$

which is at its turn equivalent to:

$$\kappa \geq \frac{[1 - \log_2 \rho(G)](1 + \eta) + \alpha_{k,L}^{(1)}[2 \log_2 \rho(G) - 1]}{4 \log_2 \rho(G) - 3} > 0.76(1 + \eta) + 2.53\alpha_{k,L}^{(1)}. \tag{6.42}$$

As $\alpha_{k,L}^{(1)} \geq 0$ and $\eta \geq 0$, it is impossible for both inequations to hold simultaneously. Thus, the first code is more efficient than the second one also with increasing inductive coupling. We can conclude that there is a higher potential in the first type of coding for performance schemes than in the second one. Moreover, it is of little interest to analyze state coding schemes that try to obtain a higher speed increase factor, as the resulting number of states decreases dramatically compared to the aforementioned ones.

In order to make sense to be employed, the first encoding scheme must assure a total throughput increase rate greater than one, i.e.:

$$\frac{1 + 4\kappa - 2\alpha_{k,L}^{(1)} + \eta}{1 + 2\kappa + \eta} \cdot \log_2 \varphi \geq 1. \tag{6.43}$$

This inequation can be reduced in the same manner as the previous ones to the following inequality:

$$\kappa \geq \frac{(1-\log_2\varphi)(1+\eta)+2\alpha_{k,L}^{(1)}\log_2\varphi}{2(2\log_2\varphi-1)} > 0.39(1+\eta)+1.78\alpha_{k,L}^{(1)}. \tag{6.44}$$

Ineq. (6.44) clearly indicates that with increasing inductive effects (growing η and $\alpha_{k,L}^{(1)}$), the encoding scheme adds up only for higher bus aspect ratios than in the purely capacitive case. It is to be noticed, that for the abovementioned set of inequalities to be correct, the condition of disjoint delay classes has to be satisfied, i.e. $\kappa \geq 2\eta + \alpha_{k,L}^{(1)}$.

6.2.2 Limits and Bounds for Transition Coding

As previously mentioned, in order to compute the exact bit rate reduction factor for transition coding, one has to determine the capacity of the constrained channel that prohibits high-delay transitions. The capacity is given by the logarithm of the spectrum of the adjacency matrix. This means that for every n, the adjacency matrix is required. The adjacency matrix for $n = 4$ and $k = 3$ can be extracted from **Tab**. 6.2 and for $n = 3$ and $k = 3$, we have:

$$A_{G3} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}. \tag{6.45}$$

It can be observed that with respect to state coding, the adjacency matrix exhibits a higher number of ones and therefore the bit rate reduction factor increases from 0.86165 to 0.98610, which is equivalent to more than 12.5 %. For $n = 4$, the increase is about 15.4 % from 0.83048 to 0.98165. **Tab**. 6.3 and **Tab**. 6.4 give the bit rate reduction factor for buses considering margins (isolated) and ignoring margins (non-isolated), respectively. The bus width varies from 2 to 12.

At this point, several properties of the bit rate reduction factor can be examined. First, the values of $\zeta_b(n,k)$ converge fast, especially when more delay classes are allowed and secondly, allowing only the delay class Δ_1 makes no sense as the number of permitted transitions decreases dramatically and thus, the spectrum of the adjacency matrix. In addition, all the bit reduction rate factors decrease with increasing n, but the one when the bus is isolated and the highest allowed delay class is Δ_1. This is due to the fact that because of the margins that have a fixed value, at low bus widths a high percentage of the

Tab. 6.3: Bit rate increasing factor for transition coding in a non-isolated bus

n	2	3	4	5	6	7	8	9	10	11	12
$\zeta_b(n,5)$	1	1	1	1	1	1	1	1	1	1	1
$\zeta_b(n,4)$	1	0.986	0.982	0.979	0.977	0.975	0.974	0.973	0.972	0.972	0.972
$\zeta_b(n,3)$	1	0.930	0.914	0.903	0.896	0.892	0.888	0.885	0.883	0.881	0.880
$\zeta_b(n,2)$	0.916	0.818	0.750	0.716	0.696	0.681	0.670	0.661	0.653	0.648	0.643
$\zeta_b(n,1)$	0.500	0.334	0.250	0.200	0.167	0.143	0.125	0.111	0.100	0.091	0.084

transitions are prohibited. However, with increasing bus width, the effect of the margins is alleviated.

Consider a 12-bit wide not isolated bus and a very high aspect ratio, i.e. $\kappa \gg 1$. Thus, the total throughput increase rate for $k = \overline{1,3}$ are:

$$\begin{aligned} \zeta_t(12,3) &= 1.295, \\ \zeta_t(12,2) &= 1.759, \\ \zeta_t(12,1) &= 2.571. \end{aligned}$$

Consequently, for high aspect ratios, encoding schemes can theoretically sped up data transmission even by 2.5 times. For $\kappa = 1$:

$$\begin{aligned} \zeta_t(12,3) &= 1.214, \\ \zeta_t(12,2) &= 1.466, \\ \zeta_t(12,1) &= 1.606. \end{aligned}$$

The computation of the spectrum is practically not feasible for high values of n. Therefore, instead of determining the highest eigenvalue, one can focus on deriving bounds for the spectrum. In the following, we analyze and improve the bounds proposed by Sotiriadis in [175].

In order to determine a practical lower bound for the spectrum we can make use of the symmetry of the adjacency matrix. For any $N \times N$ symmetric matrix A_G its spectrum $\rho(A_G)$ satisfies the inequality [175]:

$$\rho(A_G) \geq \frac{1}{N} \sum_{i=1}^{n} \sum_{j=1}^{n} A_G[i,j]. \tag{6.46}$$

Because A_G is a 0-1 matrix, the lower bound becomes for $N = 2^n$:

$$\rho(A_G) \geq \frac{p_t(n)}{2^n}. \tag{6.47}$$

where $p_t(n)$ represents the number of permitted transitions in an n-bit wide bus. Moreover, the number of transitions gives also an upper bound for the spectrum of a 0-1 matrix [175]:

$$\rho(A_G) \leq 1 + \sqrt{2p_t(n)}. \tag{6.48}$$

Tab. 6.4: Bit rate increasing factor for transition coding in an isolated bus

n	2	3	4	5	6	7	8	9	10	11	12
$\zeta_b(n,5)$	1	1	1	1	1	1	1	1	1	1	1
$\zeta_b(n,4)$	1	0.986	0.982	0.979	0.977	0.975	0.974	0.973	0.972	0.972	0.972
$\zeta_b(n,3)$	0.916	0.984	0.887	0.882	0.879	0.877	0.875	0.874	0.873	0.872	0.872
$\zeta_b(n,2)$	0.500	0.528	0.535	0.543	0.552	0.558	0.562	0.564	0.567	0.569	0.570
$\zeta_b(n,1)$	0	0	0	0	0	0	0	0	0	0	0

Nevertheless, the abovementioned upper bound is very loose especially for low and medium bus widths. It can be easily shown that for a 0-1 adjacency matrix like A_G the following inequality holds:

$$\rho(A_G) \leq \sqrt{p_t(n)}. \tag{6.49}$$

Simply put, for a given number of edges the capacity of an unweighted graph is maximized if the number of vertices is minimized. Actually, for a fixed values of ones, the highest spectrum of a 0-1 graph matrix is obtained when the ones are grouped and the abovementioned limit is reached only if the square root of the transitions number is a natural number. **Ineq**. (6.48) is in general much looser than **Ineq**. (6.49). However, in the case of extremely wide buses, the two upper bounds converge to the same value as shown later. Furthermore, bounds for the spectra of adjacency matrices can be determined in more general or particular cases like described in [23, 35, 47, 48, 49, 84, 102, 174].

Hence, in order to be able to determine the abovementioned bounds, one has to calculate first the number of allowed transitions. As for state coding, $p_t(n)$ can be computed in a recurrent manner. The delay class of a certain transitions is defined as the maximum delay in all lines and the delay in one line is a function only of itself and the first-order neighbors. So, in order to be able to find out the total number of permitted transitions at step $n+1$, the fundamental idea is to know the delay in line $n+1$ and to determine how the delay in line n is changed by adding one more bit. For this purpose, one needs to know the number of permitted transitions of all possible types at step n. The transition types at step n depend on the bit-level transitions in lines $n-1$ and n. This means that the generating or recurrence matrix T_k ($k+1$ indicates the highest allowed delay class) is a 16×16 matrix that indicates which allowed transition types can emerge at step $n+1$ from each transition type at step n by adding one more line. The generating matrices are thus easily computable non-symmetrical 0-1 matrices. As the recurrence is starting with $n=2$, there have to be defined some initial allowed transitions as the sum of the entries in the generating matrix does not give the exact transition number. These initial allowed transitions depend on the transitions in the first two bits and they differ for isolated or non-isolated buses. These conditions translate into the fact that the transition matrix has to be multiplied with a left and a right vector that indicate if transitions are prohibited when $n=2$. Consequently,

$$p_t(n) = \underline{w}_{k,l}^t T_k^{n-2} \underline{w}_{k,r}, \tag{6.50}$$

where $\underline{w}_{k,l}$ and $\underline{w}_{k,r}$ are the left and right vectors corresponding to the initial conditions, respectively.

Let P be a matrix of eigenvectors of a given square matrix T and let D be a diagonal matrix with the corresponding eigenvalues on the diagonal. The Eigendecomposition Theorem [5] says that as long as P is a square matrix, T can be written as:

$$T = PDP^{-1}. \tag{6.51}$$

Furthermore, if T is symmetric, then the columns of P are orthogonal vectors. Thus,

$$\begin{aligned} p_t(n) &= \underline{w}_{k,l}^t P_k D_k^{n-2} P_k^{-1} \underline{w}_{k,r} \\ &= \sum_{j=1}^{16} \underline{\gamma}_k[j] \cdot \underline{\lambda}_k[j]^{n-2}, \end{aligned} \tag{6.52}$$

where:

$$\underline{\gamma}_k[j] = \sum_{i=1}^{16} \underline{w}_{k,l} P_k[i,j] \cdot \sum_{i=1}^{16} \underline{w}_{k,r} P_k^{-1}[j,i], \tag{6.53}$$

and $\underline{\lambda}_k$, D_k, and P_k, represent the eigenvalue vector, the eigenvalue diagonal matrix, and the eigenvector matrix, respectively. Therefore,

$$\frac{\log_2\left(\sum_{j=1}^{16} \underline{\gamma}_k[j] \cdot \underline{\lambda}_k[j]^{n-2}\right)}{n} - 1 \leq \zeta_b(n,k) \leq \frac{\log_2\left(\sum_{j=1}^{16} \underline{\gamma}_k[j] \cdot \underline{\lambda}_k[j]^{n-2}\right)}{2n}. \tag{6.54}$$

As a result of **Eq**. (6.47) and **Eq**. (6.54) the following lower and upper bounds for bit rate reduction factor in very large buses are obtained:

$$\log_2 \rho(T_k) - 1 \leq \lim_{n\to\infty} \zeta_b(n,k) \leq \frac{\rho(T_k)}{2}. \tag{6.55}$$

For $k = \overline{0,4}$ the bounds are:

$$\begin{aligned} 1 \leq &\lim_{n\to\infty} \zeta_b(n,4) &&\leq 1, \\ 0.96369 \leq &\lim_{n\to\infty} \zeta_b(n,3) &&\leq 0.98184, \\ 0.84549 \leq &\lim_{n\to\infty} \zeta_b(n,2) &&\leq 0.92274, \\ 0.48353 \leq &\lim_{n\to\infty} \zeta_b(n,1) &&\leq 0.74177, \\ 0 \leq &\lim_{n\to\infty} \zeta_b(n,0) &&\leq 0.5. \end{aligned} \tag{6.56}$$

Nevertheless, for $k = 0$, the difference between the bounds is extreme and especially the upper bound is very loose. The bounds are very close for $k = 3$, however with decreasing allowed delay classes, the difference between the bounds increases. It is obvious that $\zeta_s(n,4) = 1$, $(\forall) n \geq 2$ and $\lim_{n\to\infty} \zeta_s(n,0) = 0$. However, by employing the properties of the bit rate reduction factor, the other bounds can be improved in order to obtain more practical ones.

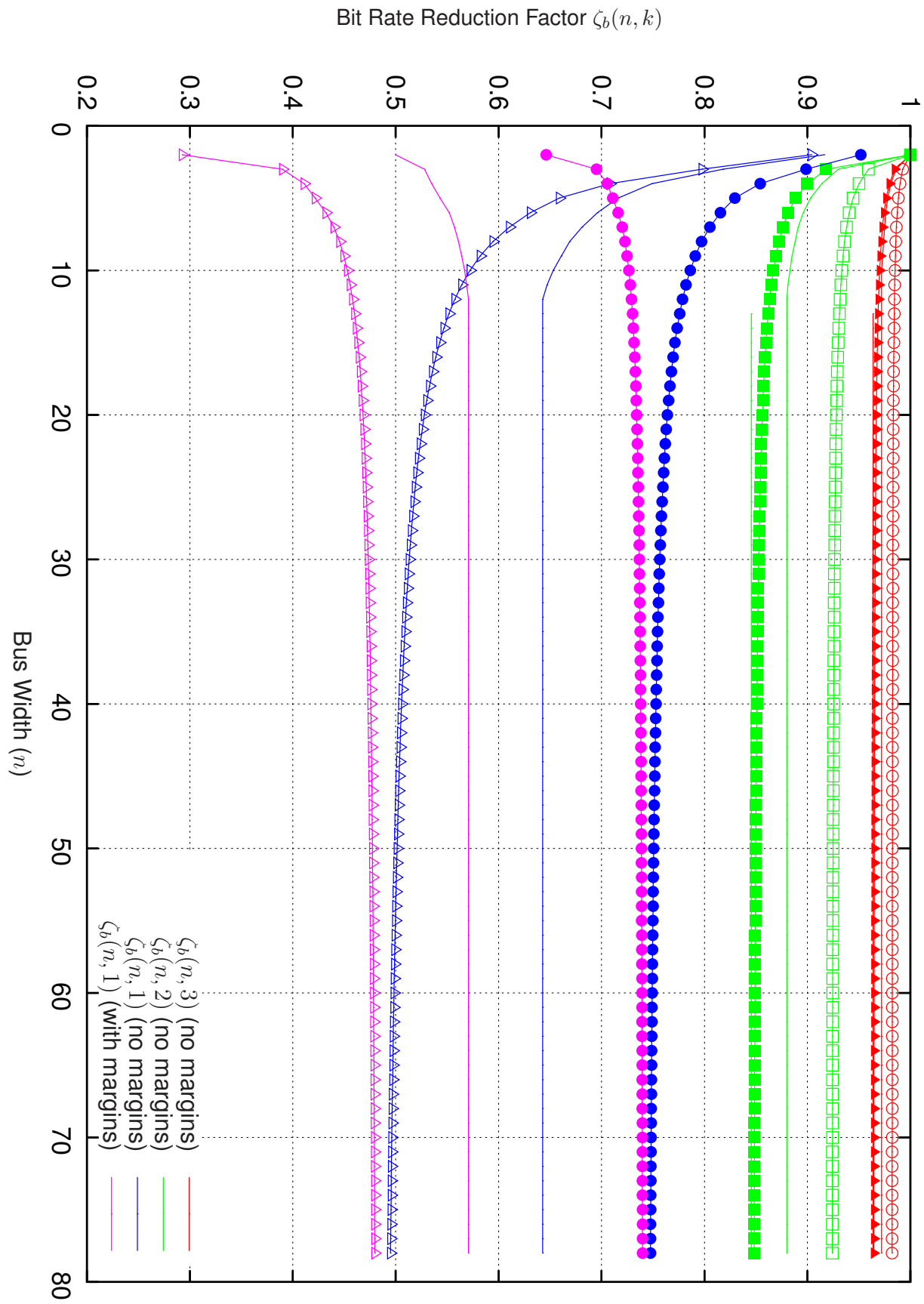


Fig. 6.3: Bounds for bit rate reduction factor

For a given bus width, the bit rate reduction factor is always greater or equal when taking into account the bus margins than when ignoring them. Moreover, as previously mentioned, $\zeta_b(n,k)$ is a continuously decreasing function of n but for $k = 1$ when the margins are not ignored. In this case, $\zeta_b(n,k)$ is a strictly increasing function. Therefore, every time the exact value of the bit rate reduction factor is calculated for a higher n, new bounds are obtained. For instance, as the exact values of $\zeta_b(n,k)$ have been computed in this work until $n = 12$, the practical bounds are for $n \geq 12$:

$$\begin{aligned} 0.96369 \leq \ & \zeta_b(n,3) \ \leq 0.97189, \\ 0.84549 \leq \ & \zeta_b(n,2) \ \leq 0.87997, \\ 0.57075 \leq \ & \zeta_b(n,1) \ \leq 0.64274, \end{aligned}$$

which are much closer than the previous ones. These practical bounds can be further improved by computing the spectrum of the adjacency matrix for higher values of n.

6.3 Coding for Throughput and Classic Anti-Crosstalk Techniques

Spacing and shielding are used in order to reduce the coupling transition activity and thus crosstalk, delay, and power dissipation. Even though very simple, both spacing and shielding belong to the generalized space of encoding schemes. In this section, simple coding schemes are described, analyzed, and combined with spacing and shielding, i.e. lower level techniques.

6.3.1 Simple Coding Schemes for Throughput

In [86] and [175], Konstantakopoulos and Sotiriadis introduced the so-called differential RLL(1,∞) – or simply D-RLL(1,∞) – coding scheme. The main idea is to employ the same decorrelator proposed for reducing power consumption (see **Chap**. 5). That decorrelator is actually reducing the two-dimensional problem of mapping the most probable transitions to the least power-hungry ones to the one-dimensional problem of reducing the number of ones in the code words.

In the case of coding for performance, no bus line exhibits a delay belonging to a class higher than Δ_3, i.e. $1 + 2\kappa$ if every vector at the output of the static mapper does not consists of successive ones. This rule defines the so-called RLL(1,∞) codes. Let Φ_n be the set of possible codewords for bus width n. By employing the same methods used in the previous section, it can be shown that the number of elements of Φ_n, is directly related to the Fibonacci sequence:

$$|\Phi_n| = F_{n+2}, \tag{6.57}$$

where $|\Phi_n|$ represents the cardinality of the set Φ_n.

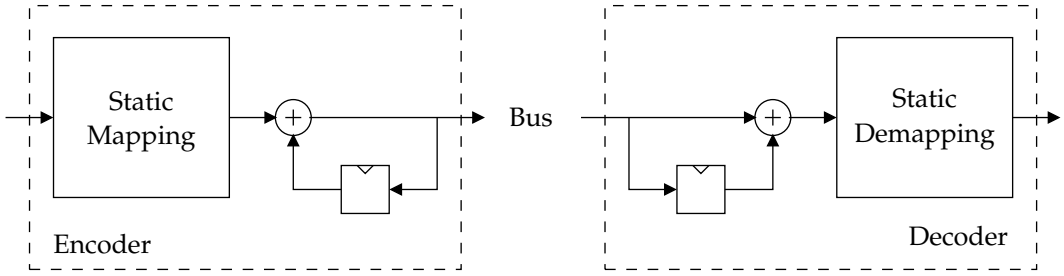


Fig. 6.4: Differential RLL(1,∞) scheme [86, 175]

Thus, the bit rate reduction factor becomes:

$$\zeta_b(n,2) = \frac{\log_2 F_{n+2}}{n} \xrightarrow[n\to\infty]{} \log_2 \varphi \simeq 0.69424, \tag{6.58}$$

and $\zeta_b(n,2)$ tends to the same value as for state coding even though for large buses, the cardinality of the codeword alphabet of the RLL(1,∞) is smaller than in the case of the state coding:

$$\frac{|\Phi_n|}{p_s(n)} = \frac{F_{n+2}}{2F_{n+1}} \xrightarrow[n\to\infty]{} \frac{\varphi}{2} \simeq 80.90\%. \tag{6.59}$$

The bit rate reduction factor can be computed also as the logarithm of the spectrum of the generating matrix:

$$G = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}. \tag{6.60}$$

The eigenvalues of G are 0, φ, and $1-\varphi$, that is $\rho(G) = \varphi$.

A very simple code avoiding Δ_5 transitions can be constructed in a similar way by not allowing the "101" sequence (or the "010" one) in any codeword of the alphabet. Thus, for a switching line, only one first-order neighbor can toggle in the opposite direction while the other aggressor remains quiet or switches in the same sense. The rule for constructing the alphabet is to introduce at least two zeros between non-adjacent ones. This code does not impose ones to be isolated in a codeword, so for this reason we call it the modified run length limited coded, M-RLL(2,∞)[2]. The codec does not require the abovementioned decorrelator-correlator structure.

The generating matrix of the code is:

$$G = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix}. \tag{6.61}$$

[2]Some authors refer to the RLL constraint as a constraint that applies only to the ones, while others refer to the same constraint for both zeros and ones

It can be shown that the eigenvalues are identical with those of the generating matrix given in **Eq**. (6.28) and that the eigenvectors of the complex eigenvalues are complex conjugated (see **Eq**. (6.35)). Thus, the cardinality of the alphabet for large buses becomes:

$$\lim_{n\to\infty} \zeta_b(n,3) \simeq 0.81137. \tag{6.62}$$

Consequently, for very large buses the M-RLL(2,∞) is less performant than the D-RLL(1,∞). For narrow buses nevertheless, due to its higher cardinality, M-RLL(2,∞) performs better than D-RLL(1,∞) if the aspect ratio is low, that is in the range:

$$0.177 \leq \kappa \leq 0.319. \tag{6.63}$$

6.3.2 Spacing and Shielding

As mentioned in **Chap**. 3, spacing (increased metal separation) and shielding are the most common crosstalk reduction techniques. In the sequel, spacing and shielding are compared without considering any form of coding and thus, the worst case delay class is always Δ_5. In the first line, inductive coupling is neglected.

If the bus width is increased by inserting r redundant bits, than with respect to the unshielded bus, the information rate remains unaltered, i.e. $\zeta_b^{(sh)} = 1$, where $\zeta_b^{(sh)}$ represents

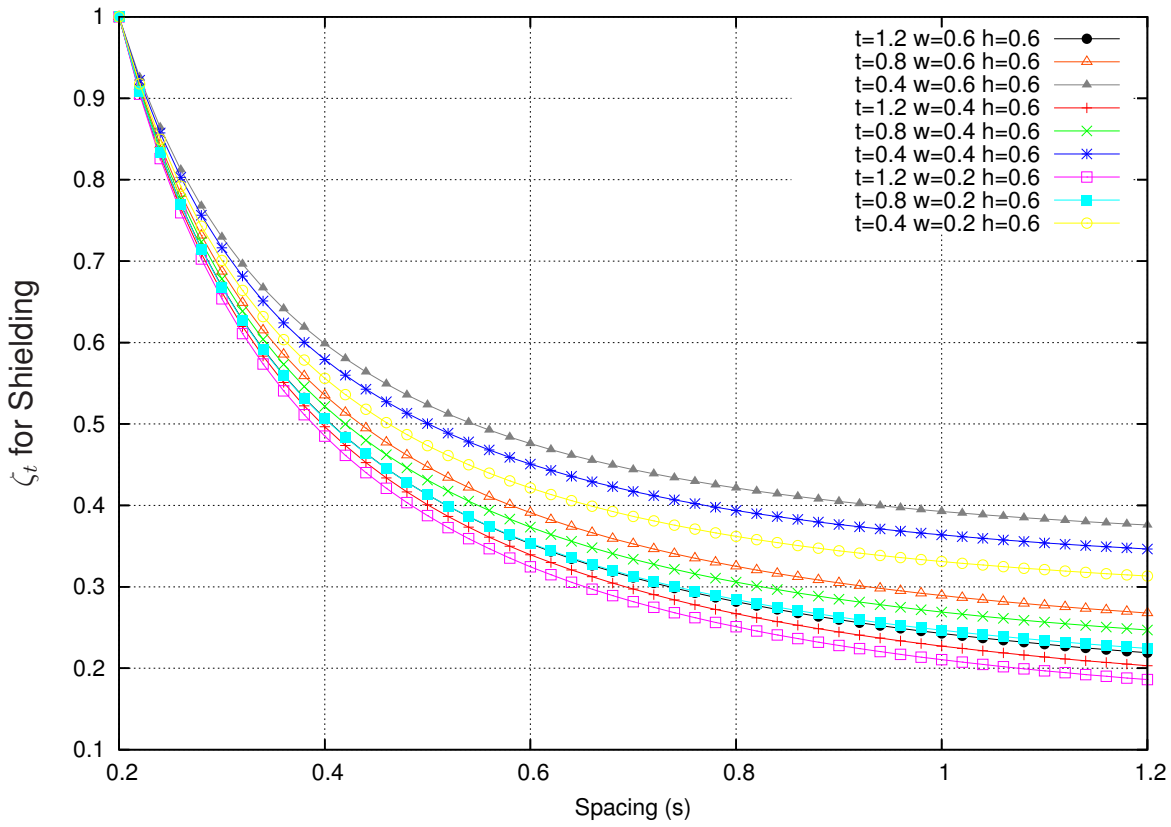


Fig. 6.5: Total throughput increase factor for shielding in capacitively coupled buses

the bit rate reduction factor. However, the speed increase factor for an uncoded shielded bus must be redefined as follows:

$$\zeta_s^{(sh)}(n,r) = \frac{C_{g,n}}{C_{g,n+r}} \cdot \frac{1+4\kappa_n}{1+4\kappa_{n+r}}, \tag{6.64}$$

where $C_{g,k}$ and κ_k represent the ground capacitance and the capacitance factor for a k-bit wide bus, respectively. Basically, there are two opposite forces: with increased spacing (decreased spacing) $C_{g,k}$ increases and $C_{c,k}$ decreases. Nevertheless, as shown also in **Fig**. 2.2 and **Fig**. 6.5 the increase rate of $C_{g,k}$ is much less aggressive than the decrease in $C_{c,k}$. In addition, shielding can be compared by considering it a coding technique applied on a bus of the same width, i.e. $n+r$. Thus, if two inserted shields span at least three signal lines, the highest allowed delay class remains Δ_5 and no delay improvement is achieved. Thus, $\zeta_b(n+r,4) = \frac{n}{n+r}$, and the total throughput increase rate is strictly less than one. Consequently, spacing is in general a more effective technique than shielding for reducing delay in capacitively coupled interconnects.

When inductive coupling is not negligible anymore, the condition for shielding to be more efficient than spacing becomes:

$$\frac{1+4\kappa_n-2\alpha_{k,L}^{(1)}+\eta}{1+4\kappa_{n+r}-2\alpha_{k,L_{sh}}^{(1)}+\eta_{sh}} > \frac{C_{g,n+r}}{C_{g,n}}, \tag{6.65}$$

where $\alpha_{k,L_{sh}}^{(1)}$ and η_{sh} are the first-order inductive coupling and the maximum cumulative inductive coupling of the aggressors of order higher than two in the shielded bus, respectively. It can be noticed that in comparison with the previous case, there are three decisive parameters that have to be taken into account. In lines that exhibit a significant amount of inductive coupling, shielding is employed in order to reduce the coupling. Therefore, low-resistance ground and V_{dd} lines are inserted between signal lines in order to provide closely spaced return paths [30,37,41,70,103,117,141,176].

When the inductive coupling is of an important amount, the high values of η can be dramatically reduced through shielding at the expense of a slightly higher $\alpha_{k,L}^{(1)}$ and an increased κ. Consequently, in inductively-coupled lines, shielding generally performs better than spacing [110].

The main disadvantage of spacing is that in order to obtain a convenient bus layout, the data characteristics have to be known at design time. If those statistics change significantly at run-time, spacing has limited applicability. However, shielding is required to minimize inductive crosstalk irrespective to data statistics. Additionally, active shielding [78,79] is an even more versatile scheme that can be combined with coding.

6.3.3 Combining Coding with Spacing

Spacing and shielding can be regarded as simple or more "primitive" coding schemes that try to reduce crosstalk effects like delay and power consumption. On the one hand,

Tab. 6.5: Combining coding and spacing ($n = 8$, $k = \overline{1,4}$, $m = \overline{2,8}$)

m	Δ_2	Δ_3	Δ_4	Δ_5
2	1.6515	1.7634	1.7259	1.6900
3	2.2890	2.3740	2.3143	2.1720
4	2.7307	2.7328	2.4885	2.2003
5	2.9917	2.7515	2.3448	1.9747
6	2.9974	2.5177	2.0357	1.6569
7	2.7362	2.1335	1.6616	1.3227
8	2.2710	1.6763	1.2737	1

increasing line spacing implies a reduction in bit rate per cycle and can be regarded as a serialization of the data transmission, and on the other hand, shielding is equivalent to introducing redundancy at a constant total bit rate. Moreover, coding can be combined with lower level techniques like shielding and/or spacing in order to achieve even a higher total throughput increase rate. In the following, the suitability of combining spacing with coding is analyzed.

Let n, k, and m denote the initial bus width, the maximum allowed delay class, and the resulting bus width, respectively. Thus, the bit reduction rate and the speed increase factor can be defined in a more general fashion:

$$\zeta_b(n,k,m) \stackrel{\text{def}}{=} \frac{m}{n} \cdot \zeta_b(m,k) \quad \text{and} \tag{6.66}$$

$$\begin{aligned} \zeta_s(n,k,m) &\stackrel{\text{def}}{=} \frac{C_{g,n} + 4C_{c,n}}{C_{g,m} + kC_{c,m}} \\ &= \frac{C_{g,n}}{C_{g,m}} \cdot \frac{1 + 4\kappa_n}{1 + k\kappa_m} \\ &= \frac{C_{g,n}}{C_{g,m}} \cdot \frac{1 + 4\kappa_n}{1 + 4\kappa_m} \cdot \zeta_s(m,k), \end{aligned} \tag{6.67}$$

where

$$\zeta_b(n,k) \stackrel{\text{def}}{=} \zeta_b(n,k,n), \quad \text{and} \tag{6.68}$$

$$\zeta_s(n,k) \stackrel{\text{def}}{=} \zeta_s(n,k,n). \tag{6.69}$$

Thus, the total throughput increase rate becomes:

$$\zeta_t(n,k,m) = \frac{m}{n} \cdot \frac{C_{g,n}}{C_{g,m}} \cdot \frac{1 + 4\kappa_n}{1 + 4\kappa_m} \cdot \zeta_t(m,k). \tag{6.70}$$

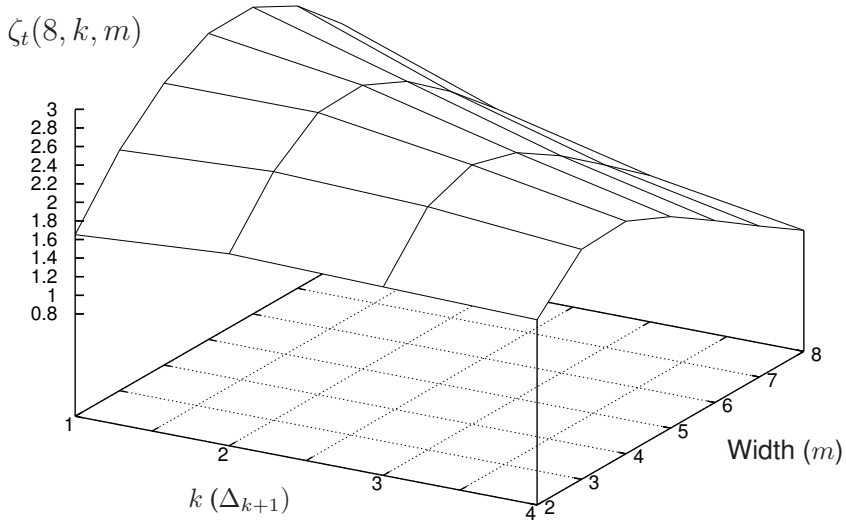


Fig. 6.6: Combining coding and spacing. Total throughput increase factor

If the bus width is fixed, the total throughput increase rate can be written as:

$$\zeta_t(n, k) \stackrel{\text{def}}{=} \zeta_t(n, k, n). \tag{6.71}$$

Tab. 6.5 and **Tab**. 6.6 illustrate the theoretical total throughput increase rate for several hybrid schemes consisting of coding and spacing when applied on an eight-bit wide bus with $t = 1.2$ µm, $w = 0.4$ µm, and $h = 0.6$ µm. It can be observed that for the described bus geometry, coding (the last row) performs theoretically slightly better than pure spacing (last column).

Nevertheless, when combining coding and spacing the maximum achievable throughput increase rate augments by more than 30 %. It is to be noticed that finding the best spacing is equivalent to finding the optimal compromise between serial and parallel data transmission. Basically, for every fixed k, the goal is to find the m that maximizes the achievable total throughput increase rate.

6.4 Simultaneous Power and Performance Optimization

As shown in the previous chapters and sections, coding can be employed in order to improve performance and power consumption. However, encoding schemes can be combined and enhanced for simultaneously optimizing performance and power. This section discusses theoretical aspects related to simultaneous power and performance improvement and illustrates ways to construct hybrid schemes.

6.4.1 Relating Delay and Transition Activity

In [175], it has been shown that the average line delay, δ_m, is directly proportional to the average dissipated energy per line. In terms of equivalent transition activity, the average delay in an N-bit wide bus is:

$$\begin{aligned}
\delta_m &= \mathbf{E}[\delta_i] = \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{N} \delta_i \\
&= \lim_{n\to\infty} \frac{\tau_0}{n} \sum_{i=1}^{N} \Delta b_i \big[(1+2\kappa)\Delta b_i - \kappa(\Delta b_{i+1} + \Delta b_{i-1})\big] \\
&= \tau_0 \cdot \lim_{n\to\infty} \left[\frac{1}{n} \sum_{i=1}^{N} \Delta b_i^2 + \frac{\kappa}{n} \sum_{i=1}^{N} \Delta b_i (2\Delta b_i - \Delta b_{i+1} - \Delta b_{i-1}) \right].
\end{aligned} \tag{6.72}$$

Let $t_{eq,m}$ be the mean equivalent transition activity:

$$t_{eq,m} = \mathbf{E}[t_{eq,i}] = \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{N} t_{eq,i}, \tag{6.73}$$

where $t_{eq,i}$ is the equivalent transition activity in line i. It can be easily shown that:

$$\delta_m = 2\tau_0 t_{eq,i}. \tag{6.74}$$

The average line delay is thus directly proportional to the equivalent transition activity.

An immediate observation is that by reducing the mean delay, the transition activity is also decreased and vice-versa. Nonetheless, reducing the mean delay is not necessarily equivalent to a decrease of the worst case delay. On the contrary, as indicated later in **Sec**. 6.4.3, power consumption can be dramatically decreased by spacing lines with small coupling activity very close to one another and those with a high activity as distant a possible. However, during this process the effects on the worst case delay are ignored. Therefore, in order to optimize delay and power simultaneously or to manage performance versus power, constraints for the worst case delay are required. One can optimize power consumption under those constraints or other figure of merits like the power-delay product (PDP).

6.4.2 Optimizing Delay and Self Transition Activity

Konstantakopoulos developed in [86] a scheme that implements a version of the aforementioned D-RLL(1,∞) code. A 4-bit wide bus is extended to 6 bits and the encoder maps the input data to symbols that do not have adjacent bits equal to one. Thus, the highest allowed delay class is Δ_3. It is to be noticed that as discussed in **Sec**. 6.3.3, the effectivity of such an implementation that expands the bus depends on the resulting increased bus aspect ratio. The scheme is efficient only if:

$$\frac{C_{g,4}}{C_{g,6}} \cdot \frac{1+4\kappa_4}{1+2\kappa_6} \geq 1. \tag{6.75}$$

The goal was to design an encoder consisting out of as few as possible gates. For this purpose, one input bit has been hardwired directly to an output one. This requires the use of at least four symbols of weight three. Nevertheless, the encoder can also be designed to minimize the self activity on the bus, i.e. for reducing the mean symbol weight [15, 143, 144]. **Tab**. 6.6 illustrates one of the many possible implementations of a codec characterized by codewords with weights not larger than two.

The total self transition activity depends on the statistical data characteristics. For simplicity, uniformly distributed data is considered. Thus, the mean symbol weight is reduced from 1.75 to 1.5. This corresponds to an improvement of about 14.28 % in total self activity with respect to the scheme developed in [86].

Tab. 6.6: D-RLL(1,∞) implementation for minimal self transition activity

$b_{in,3}$	$b_{in,2}$	$b_{in,1}$	$b_{in,0}$	$b_{out,5}$	$b_{out,4}$	$b_{out,3}$	$b_{out,2}$	$b_{out,1}$	$b_{out,0}$
0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	1
0	0	1	0	0	0	0	0	1	0
0	0	1	1	0	1	0	0	0	0
0	1	0	0	0	0	0	1	0	0
0	1	0	1	0	0	0	1	0	1
0	1	1	0	1	0	0	0	0	1
0	1	1	1	1	0	0	0	0	0
1	0	0	0	0	0	1	0	0	0
1	0	0	1	0	0	1	0	0	1
1	0	1	0	0	0	1	0	1	0
1	0	1	1	1	0	0	0	1	0
1	1	0	0	1	0	0	1	0	0
1	1	0	1	0	1	0	1	0	0
1	1	1	0	0	1	0	0	1	0
1	1	1	1	1	0	1	0	0	0

6.4.3 Optimizing Delay and Total Transition Activity

The scheme developed for the D-RLL(1,∞) code could also be improved to additionally reduce the coupling transition activity or the total equivalent coupling activity. The problem formulation is to find for an n-bit wide bus the so-called power optimal mapping function (code) Ψ_C among all mapping functions such that the average energy consumption is minimized. The average energy consumption, E_m, is defined for a code Ψ_C as follows:

$$E_m = \sum_{i=1}^{n}\sum_{j=1}^{n} p(\underline{b}_i, \underline{b}_j)\cdot E\big(\Psi_C(\underline{b}_i), \Psi_C(\underline{b}_j)\big) \tag{6.76}$$

where $p(\underline{b}_i, \underline{b}_j)$ represents the transition probability from $\underline{b}_i$ to $\underline{b}_j$, and $E\big(\Psi_C(\underline{b}_i), \Psi_C(\underline{b}_j)\big)$ is the energy associated to the toggling from $\Psi_C(\underline{b}_i)$ to $\Psi_C(\underline{b}_j)$.

For constant bit rate per symbol in an n-bit wide bus, the cardinality of a D-RLL(1,∞) alphabet is F_{m+2}, where $\log_2 F_{m+2} \geq n$. Therefore, the total number of possible mapping functions is given by:

$$\mathrm{P}^{2^n}_{F_{m+2}} = \frac{F_{m+2}!}{(F_{m+2}-2^n)!} = 2^n!\cdot\binom{F_{m+2}}{2^n} \tag{6.77}$$

where $\mathrm{P}^k_n = \frac{n!}{(n-k)!}$ and represents the number of permutations of n different things taken k at a time. For $n = 8$, the first Fibonacci number greater than $2^8 = 256$ is $F_{14} = 377$. Thus, $m = 12$ and the total number of possible codes is $\mathrm{P}^{256}_{377} = \frac{377!}{121!}$. Consequently, it is virtually impossible to search for the best code in an exhaustive manner even for narrow buses.

Another efficient way to reduce power consumption in a bus when the data statistics are known *a priori* to the design is asymmetrical spacing. In this way, neighboring lines exhibiting a high coupling activity are more widely spaced than those with a low coupling activity. Actually, asymmetrical spacing is a technique that trades power for performance by finding the set of spacings $s_{i,i+1}$ for $i = \overline{1, n-1}$ that minimizes the weighted coupling transition activity, T_{Cw}:

$$T_{Cw} = \sum_{i=1}^{n-1} t_c(i,i+1)\cdot\kappa(i,i+1) = \sum_{i=1}^{n-1} t_c(i,i+1)\cdot\kappa(s_{i,i+1}) \quad \text{with} \tag{6.78}$$

$$\sum_{i=1}^{n-1} s_{i,i+1} = \text{constant} \tag{6.79}$$

Tab. 6.7 shows the optimal spacings calculated with a branch-and-bound algorithm for a synthetic 8-bit signal with μ=0, σ_n=0.19531, and ρ=0.930. Thickness, height, and width have been set to 1.2 µm, 0.6 µm, and 0.4 µm, respectively. The minimum permitted spacing has been varied between 0.02 µm and 0.018 µm. It can be observed that while the power consumption decreases at a significant rate, the bus aspect factor increases much more rapidly for this type of bus geometry and coupling activity.

Asymmetrical spacing can achieve a significant reduction of the total transition activity, however at the expense of an important performance loss. Contrary to coding

Tab. 6.7: Normalized power and κ for different minimum spacings

s_{min} [µm]	Norm. Power	κ_{max}	$s_{1,2}$	$s_{2,3}$	$s_{3,4}$	$s_{4,5}$	$s_{5,6}$	$s_{6,7}$	$s_{7,8}$
0.18	93.94 %	4.59	0.220	0.220	0.210	0.200	0.180	0.180	0.190
0.16	88.20 %	5.43	0.230	0.230	0.230	0.220	0.160	0.160	0.170
0.14	64.86 %	6.55	0.240	0.240	0.240	0.220	0.160	0.140	0.160
0.12	80.14 %	8.09	0.280	0.280	0.220	0.220	0.160	0.120	0.120
0.10	76.80 %	10.27	0.280	0.280	0.220	0.220	0.200	0.100	0.100
0.08	72.83 %	13.60	0.280	0.280	0.240	0.240	0.200	0.080	0.080
0.06	70.80 %	19.17	0.275	0.275	0.275	0.260	0.185	0.060	0.070
0.04	69.67 %	21.19	0.275	0.275	0.275	0.265	0.190	0.065	0.055
0.02	68.42 %	30.22	0.280	0.280	0.280	0.260	0.200	0.060	0.040

which can be employed at run-time, this technique is applicable only at design time if the data statistics are known *a priori*. At run-time, instead of asymmetrical spacing, one can implement an active shielding scheme whenever the bus width is greater than the number of bits required for data representation. For instance, one bus line remains unused when a 7-bit wide signal is mapped on an 8-bit fixed wide bus. In order to reduce power consumption, the unused bit should be mapped between the bits exhibiting the highest coupling activity. The effectiveness of the shielding depends actually on the exact coupling activity.

Tab. 6.8 shows the coupling activity in the case of non-isolated non-shielded and quietly shielded 2-bit uncorrelated and uniformly distributed data. It can be easily shown that the coupling activity with a shield inserted between the two signal lines is equal to the self activity and thus, T_c=0.5 in both cases. Consequently, for uncorrelated uniformly distributed 2-bit data the quiet shield does not bring along any advantage. In order to reduce the coupling activity between b_0 and b_1, the following coding function that expands a bus from n to n+1 bits can be defined:

$$\Psi_C([b_{n-1}, b_{n-2}, \ldots, b_0]) = [b_{n-1}, b_{n-2}, \ldots, b_1, b_{sh}, b_0], \tag{6.80}$$

where the shielding line is computed as:

$$b_{sh} = b_0^+ b_1^+ (b_0^- b_1^- + \overline{b_0^-}\,\overline{b_1^-}) = b_0^+ b_1^+ (b_0^- \oplus \overline{b_1^-}). \tag{6.81}$$

Tab. 6.8: Bit coupling activity for unshielded and shielded 2-bit data

	00	01	10	11
00	0	1	1	0
01	0	0	2	0
10	0	2	0	0
11	0	1	1	0

	00	01	10	11
00	0	1	1	2
01	0	0	1	1
10	0	1	0	1
11	0	0	0	0

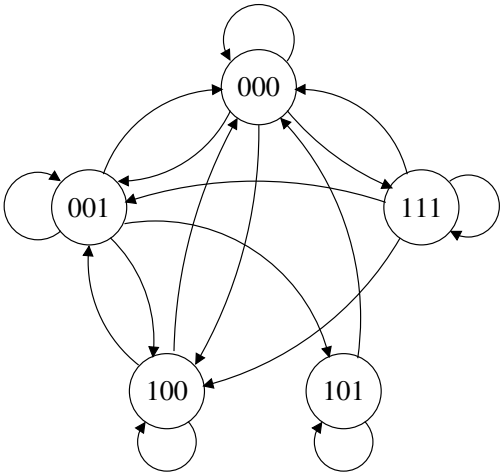


Fig. 6.7: Markov process describing the coupling activity in a shielded 2-bit bus

The transitions in the bits b_1, b_{sh}, and b_0 are equivalent to the stochastic process represented in **Fig**. 6.7. The edges have equal probability and therefore, the associated stochastic matrix is:

$$P = \frac{1}{4} \cdot A_G = \frac{1}{4} \cdot \begin{pmatrix} 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \end{pmatrix}, \tag{6.82}$$

and the state probability vector (the fixed vector of P – see **App**. B) is:

$$w = \frac{1}{12} \cdot \begin{bmatrix} 3 & 3 & 3 & 2 & 1 \end{bmatrix}^t. \tag{6.83}$$

Thus, the resulting coupling activity is $T_c = 3/8$. The cost for the reduction in coupling activity is a small self transition activity in the shielding line, i.e. $t_{s,sh} = 1/16$.

When combining coding for performance and coding for power into a single scheme, one has to apply first the coding for speed. After the highest tolerated delay class is defined, the redundancy can be used for further reducing the transition activity if possible. As previously shown, coupling activity can also be traded for self activity.

If the entropy of the uncoded signal is less than $\frac{2n}{3}$, where n is the bus width, a version of the M-RLL(2,∞) scheme can be implemented by inserting an active shield at least after each pair of signal bits. The shielding line is defined exactly as in **Eq**. (6.81). If the entropy is less than $\frac{n}{2}$, then each signal bit can be doubled by a shielding line and a very simple version of the D-RLL(1,∞) scheme can be implemented. Further, if the entropy is even smaller than $\frac{n}{3}$, the highest delay class that must be allowed is Δ_2 as for each signal bit two redundant ones can be used. All these schemes are very effective to be used in combination with wire splitting, because during this process for each signal line, at least one split is added, which results in a redundancy that can be efficiently exploited as previously described.

6.5 Summary

Coding can be used not only for reducing power consumption but also to improve the throughput of a bus. Basically, worst case transitions are eliminated in order to speed up the bus. There are two ways to encode a bus: state coding, in which states involved in high-delay transitions are prohibited; and transition coding, where only the worst case transitions are not permitted.

First, benefits and shortcomings of both coding types have been highlighted. The developed pattern-dependent delay (ELD) model has been employed to show how the delay classes identified in capacitively coupled buses evolve with growing inductive effects. One essential contribution of this chapter is the computation of limits and bounds for both state and transition coding. In addition, limits for the bus aspect factor have been calculated in order to be able to assess the effectiveness of the encoding schemes. That following, simple coding schemes for throughput improvement are developed and evaluated. Moreover, the pattern-dependent delay model has been used also for assessing the effectivity of spacing and shielding. The delay model is well suited for choosing among the best anti-crosstalk and delay-improving techniques.

Furthermore, it has been shown how signal encoding can simultaneously improve performance and power. The mean delay has been related to the average dynamic power consumption and it has been highlighted that reducing the mean delay is not necessarily correlated to decreasing the transition activity. Nonetheless, it has been shown that in the case of differential schemes like D-RLL(1,∞), self activity can be easily reduced by assigning the codewords with minimum weight to those with the highest probability of appearance. Finally, the generalized power macromodel has been used for developing a spacing-based design-time coding and an active-shielding-like run-time coding for delay and total transition activity optimization. The combination of coding with lower level techniques like spacing and (active) shielding represent an important contribution of this work.

Chapter 7

Methodology Binding

Contents

The highly competitive environment in current integrated circuit design poses a tremendous pressure on the employed design flows, as even small and apparently insignificant variations in the quality of one flow or another can make the difference between success and failure [24]. However, it is actually impossible to clearly quantify the efficiency of any flow mainly because finding a metric for design technology is a rather intractable issue. The vast majority of design problems implicate the determination of the optimal trade-off among a multitude of design parameters and metrics like design costs, time-to-market, performance, power consumption, or architectural flexibility. As also pointed out in [24], in order to successfully cope with augmenting system complexity and design costs, stringent time-to-market requirements, as well as with upcoming tricky VDSM effects, breakthroughs via new algorithmic approaches usually manifested in tools have to be accompanied by basic changes in design problem formulations.

In very deep sub-micron technologies, interconnects become the dominant factor in performance, complexity, and power consumption [30, 37, 113, 164, 191]. Given this increasingly dominant importance of interconnects, design flows have to be adapted to accommodate interconnect analysis, synthesis, and optimization methods at every level of abstraction, especially at higher ones. The envisaged emphasis on interconnects re-

quires the integration of both new, specific algorithms and tools, and fundamentally new methodologies and associated tool flows. The necessity of addressing interconnect (and buffer planning) at early stages of the design flow has been already highlighted in the literature. For instance, Cong presented in [32] an interconnect-centric design flow for nanometer technologies (see **Chap**. 3).

The objective of this chapter is twofold. First, the importance of tackling the interconnect design issue during the very early stages of the design flow is highlighted. In this context, a simultaneous placement and buffer planning method for reduction of power consumption in interconnects and repeaters is developed. Furthermore, coding for throughput is appended to the developed algorithm in order to show the effectiveness of combining coding with lower level techniques – in this case buffer planning. The technique shows the significant optimization opportunities in terms of both performance and power consumption available during the early design phases. Secondly, a scheme for an interconnect-oriented design flow is discussed that allows a practical and seamless integration of macromodels and signal encoding schemes like those introduced throughout this thesis.

7.1 High-Level Optimization of Buffered Interconnects

Buffer insertion is a very effective and probably the most popular method to reduce interconnect delay and lately also crosstalk, by breaking long wires into shorter ones and inserting signal repeating gates. Since controlling power dissipation and density is becoming in many ways more daunting than timing closure [24], the importance of estimating and optimizing power consumption at early design stages is steadily increasing.

Traditionally, the inserted buffers barely influenced the total area and power consumption of a system. However, repeaters are reported to become a problem at both chip- and block-level [165]. The percentage of total (local and global) repeaters in a design is projected to reach 35% by the 45 nm technology node and even 70% by the 32 nm node. This means that buffers will eventually be responsible for the majority of the die area and total static power consumption (leakage-induced). The dynamic component of power consumption will be primarily determined by the total interconnect structure, i.e. switching capacitances of wires and repeaters. Such an explosion in repeater number would finally have a profound impact on the design flow. Consequently, issues like minimizing area and power consumption in buffered interconnects need to be tackled when the largest optimization opportunities are available, that is at the very early stages of the design flow [24, 30, 32, 165].

Cong showed that early interconnect planning has a tremendous impact on the final results [32]. Later, Ma et al. proposed in [106] to integrate buffer planning into a Simulated-Annealing-based floorplanning. However, the main goal in that work was to address buffer allocation early in the design flow from the perspective of performance and routing congestion. The method developed in this thesis is an extension of a classical

placement algorithm based on Simulated Annealing in order to include power-optimized interconnect and global buffer planning. The placement algorithm has been merged with van Ginneken's buffer insertion algorithm, A-Tree construction, and moment-based delay computation in RC trees.

7.1.1 Placement, Routing, and Buffer Insertion

Placement is the process of arranging the circuit components on a layout surface. Given a collection of cells or modules with ports, the dimensions of these cells, and a collection of nets, the process of placement consists of finding suitable physical locations for each cell on the entire layout. By suitable it is meant that given objective functions are to be optimized [54, 159].

Traditionally, routing information have been omitted from the symbolic placement. However, from a symbolic placement, it is possible to get an estimate of the routing requirements or the power consumed in the routing structure. In order to improve the efficiency of placement and routing, the two steps are usually performed iteratively, especially because they influence one another significantly. The complexity of the iteration loop is determined by the trade-off between time efficiency and solution optimality. During placement, several simple metrics can be used for estimating the impact of placement on routing, for instance the half-perimeter (HP) or the total tree-wirelength [54,159]. Simulated Annealing (SA) is one the most well developed and often employed iterative technique for solving several combinatorial optimization problems [84]. This adaptive heuristic has been widely used for partitioning, floorplanning, placement, etc. Its main advantage is that the cost function can be easily enhanced, and based on the obtained results the various weighting coefficients can also be adapted.

The main objective when constructing an interconnect tree is to search for the shortest path which connects all nodes. This formulation is also called the Steiner-tree problem [54, 159, 170]. However, the minimal Steiner-tree is not necessarily optimal with respect to the required arrival time (RAT). Therefore, an efficient algorithm tries to minimize the resulted delay during the tree construction. In order to simplify the computation of the delay associated to a gate and the corresponding interconnect, a two-step approximation is usually employed [26]. The total stage delay is the sum of gate and interconnect delay separately. To capture the delay by means of a simple empirical model, the RC load is replaced by an effective capacitance, C_{eff}, that accurately characterizes the interconnect delay [36, 140]. Once the gate output transition time is calculated, the output waveform is approximated with a ramp, which is afterwards used to determine the interconnect delay [26]. It is to be noticed, that during this design phase, topology, physical hierarchy, wire sizing, spacing or splitting are not known. For this reason and also in order to limit the complexity, more simple and thus less accurate design metrics are employed. In this case, delay estimators based on the first one (Elmore) or two moments (D2M) can be applied.

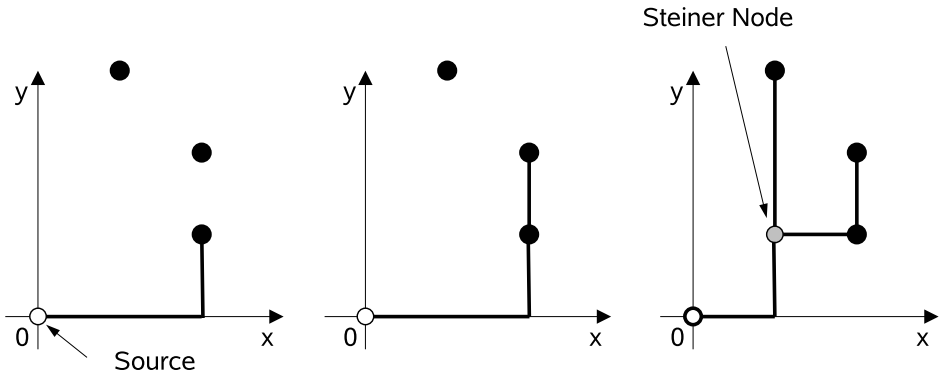


Fig. 7.1: SERT Algorithm

The SERT-Algorithm (Steiner Elmore Routing Tree) proposed in [19] is connecting only two nodes at a time. As shown in **Fig**. 7.1, the algorithm chooses at each step the sink with the smallest maximum delay (for instance the Elmore delay) in the current tree. Existing connections can be thereby deleted and so-called Steiner nodes are inserted. Another very efficient tree construction algorithm is the so-called A-Tree algorithm [34, 131, 148]. A rectangular Steiner-tree is an A-Tree if each path connecting a sink to the source represents the shortest path. Thus, an A-Tree guarantees at the expense of a larger total interconnect length smaller arrival times. As illustrated in **Fig**. 7.2, the idea behind the method is to merge sub-trees with the largest common minimal distance to the source. The two methods have been compared in [30] and it has been observed that A-Tree generally outperforms SERT.

As previously mentioned, buffer insertion is one of the most popular and effective techniques to achieve timing closure. In [195], van Ginneken introduced an algorithm which determines the optimal positions to insert buffers in RC trees. This algorithm laid the fundament for a significant amount of subsequent buffer insertion methods.

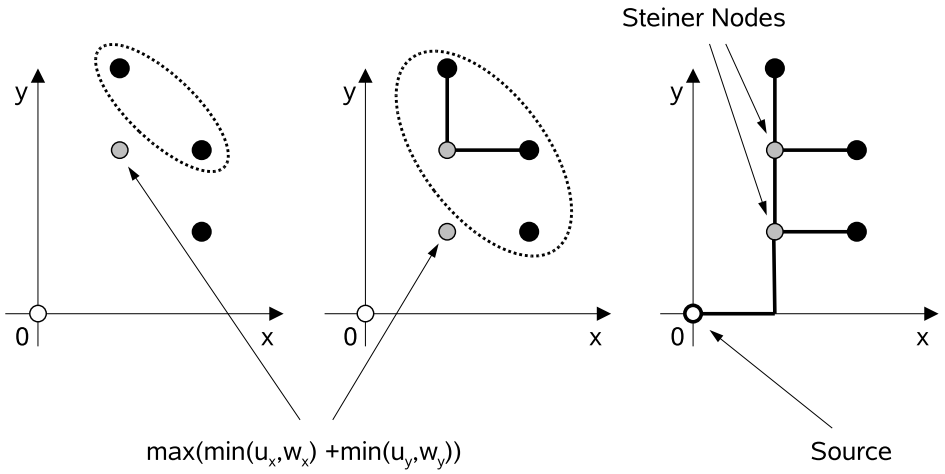


Fig. 7.2: A-Tree Algorithm

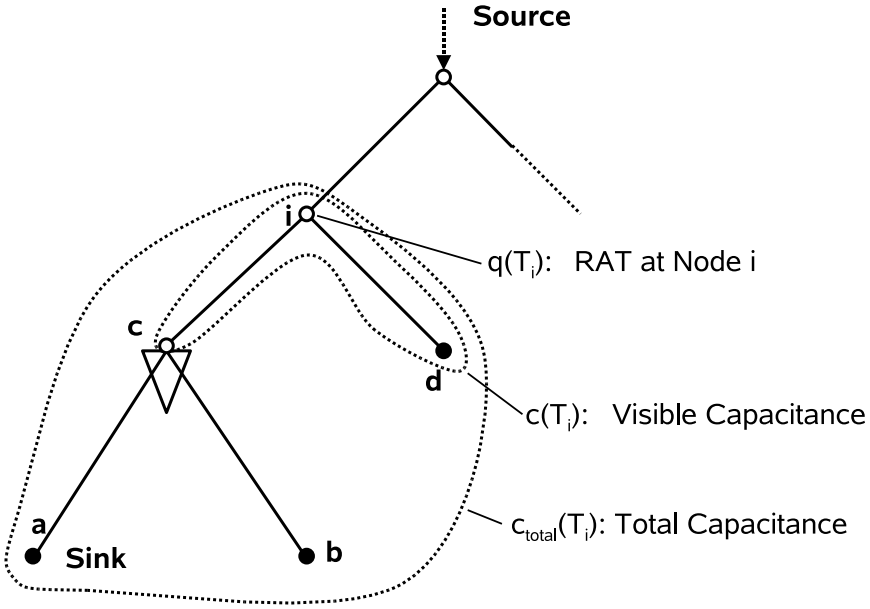


Fig. 7.3: Options in van Ginneken's Algorithm

The input of the algorithm is a net with one source and at least one sink. All pins have to be already connected through a Steiner-tree. Moreover, the possible buffer positions in the net have to be known too. The main idea of the algorithm is to compute for each possible buffer position the RAT-capacitance pair. The complexity of the problem is reduced by splitting it into smaller sub-problems (*divide et impera*) and by combining the obtained solutions. All calculations are performed from the tree bottom towards the source. The buffer insertion algorithm also requires that slight layout modifications are allowed, such that buffers can also be integrated in existing modules.

In addition, the following parameters have to be known: source output resistance, r_{out}, input capacitance of the sinks, c_i, the latest possible arrival time for each sink, q_i. The delay computation employs the aforementioned Elmore formula. The algorithm computes in a sub-tree T_i for each possible buffer position an option, i.e. the pair RAT-capacitance, $(q(T_i), c(Ti))$. **Fig**. 7.3 shows such an option for node i. The option includes a buffer that has already been inserted in node c. Therefore, the only visible nodes in i are c and d. However, the algorithm had already computed the required arrival time for the buffer during the previous step. Thus, the buffer is equivalent to any other sink with the corresponding input capacitance.

Since van Ginneken's seminal paper, an important amount of modifications to the original algorithm have been proposed. For instance, Lillis et al. proposed in [97] simultaneous buffer insertion/sizing and wire sizing (BISWS) for delay optimization in a given Steiner-tree, Chu and Wong derived in [31] elegant closed form solutions to Wire Sizing (WS), simultaneous BISWS, and simultaneous BISWS for a fixed number of buffers (BISWS-m), only to cite a few. In the last years, the issue of power optimal buffer insertion has also been addressed (see [96] for a brief overview).

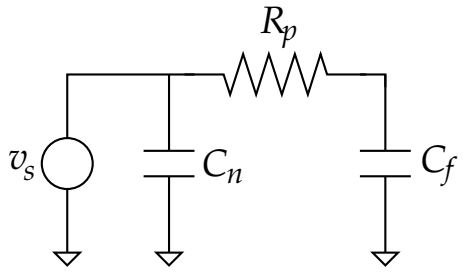


Fig. 7.4: Pi-model

With increasing interconnect density, wire resistance cannot be neglected anymore and the phenomenon of resistive shielding has to be taken into consideration. This means that because of the significant resistance at the driver output, only a portion of the total net capacitance is actually seen. Rather than analyzing the entire interconnect structure to calculate the average current that flows in at each step, one can employ a reduced-order driving point model [26, 76, 130, 140, 164]. This can be accurately achieved with a simple Pi-model like in **Fig**. 7.4.

Let $Y(s) = \sum y_n s^n$ be the driving point admittance function of the gate load, where y_i is the i-th moment of $Y(s)$. By matching the moments up to the third order, the Pi-circuit parameters can be calculated as follows: $C_f = \frac{y_2^2}{y_3}$, $C_n = y_1 - \frac{y_2^2}{y_3}$, and $R_\pi = -\frac{y_3^2}{y_2^3}$. Further, the effective capacitance is calculated as $C_{eff} = C_n + \beta C_f$, where $0 < \beta < 1$. The factor β can be

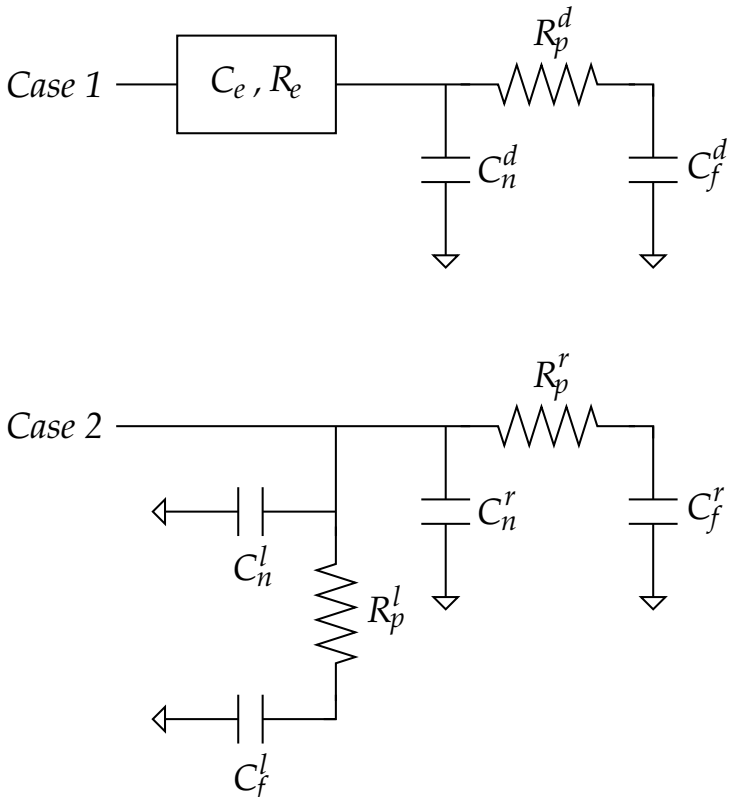


Fig. 7.5: Two merging cases for Pi-model calculation (after [3])

CALCPI$(e, \pi^d = (C_n^d, R_\pi^d, C_f^d))$
1 $y_1^d = C_n^d + C_f^d; y_2^d = -R_\pi^d (C_f^d)^2;$
2 $y_3^d = (R_\pi^d)^2 (C_f^d)^3;$
3 $y_1 = y_1^d + C_e;$
4 $y_2 = y_2^d - R_e[(y_1^d)^2 + C_e y_1^d + (C_e^2/3)];$
5 $y_3 = y_3^d - R_e[2y_1^d y_2^d + C_e y_2^d] + R_e^2[(y_1^d)^3 + \frac{4}{3} C_e^2 y_1^d + \frac{2}{15} C_e^3];$
6 **return** $\pi = (y_1 - (y_2^2/y_3), -y_3^2/y_2^3, y_2^2/y_3);$

Listing 7.1: Pi-circuit calculation in the first case [3]

iteratively computed at run-time or precomputed and saved in a look-up table or a data base [26, 164]. By employing the effective capacitance instead of the total net capacitance, the obtained delays are smaller and thus much closer to reality.

In order to achieve accurate delay computations, the set of methods introduced by Alpert et al. in [3] have been applied. The methods allow to perform the computations starting from the tree leaves, which makes it compatible to the van Ginneken algorithm. The delay estimation in a Steiner-tree based on the Pi-model is used for calculating both C_{eff} and higher-order moments. As shown in **Fig**. 7.5, two different cases need to be taken

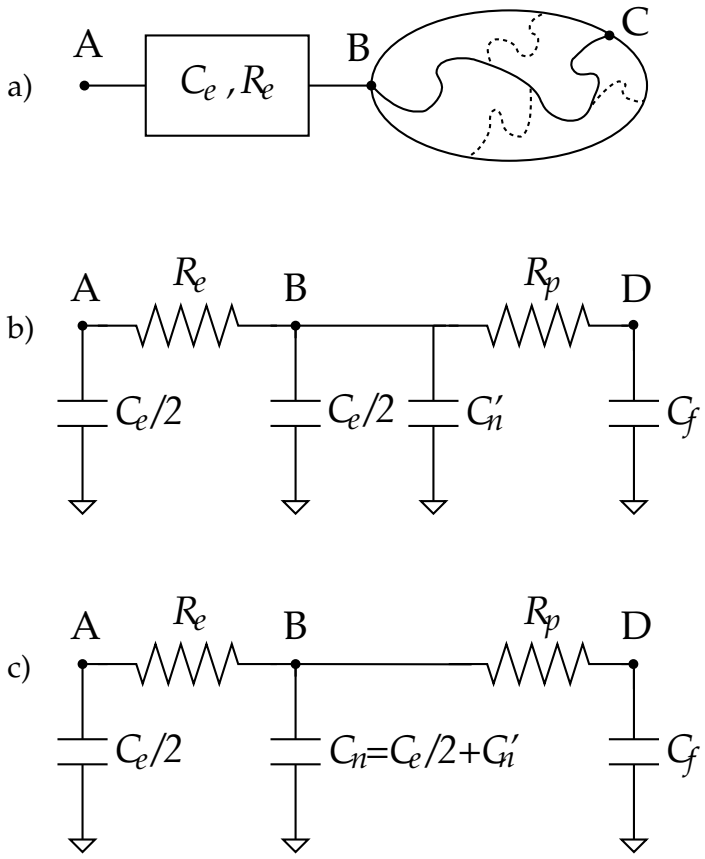


Fig. 7.6: Computation of moments (after [3])

CALCPI($\pi^l = (C_n^l, R_\pi^l, C_f^l), \pi^r = (C_n^r, R_\pi^r, C_f^r)$)
1 $y_1^l = C_n^l + C_f^l; y_1^r = C_n^r + C_f^r;$
2 $y_2^l = -R_\pi^l (C_f^l)^2; y_2^r = -R_\pi^r (C_f^r)^2;$
3 $y_3^l = (R_\pi^l)^2 (C_f^l)^3; y_3^r = (R_\pi^r)^2 (C_f^r)^3;$
4 $y_1 = y_1^l + y_1^r; y_2 = y_2^l + y_2^r; y_3 = y_3^l + y_3^r;$
5 **return** $\pi = (y_1 - (y_2^2/y_3), -y_3^2/y_2^3, y_2^2/y_3);$

Listing 7.2: Pi-circuit calculation in the second case [3].

into consideration. In the first case, an existing Pi-model is merged with a wire segment e, while in the second one, two Pi-circuits are merged together. The two functions are explained in detail in [3] and are given in Listings 7.1 and 7.2, respectively. With these two functions, it is possible to compute for each buffer position a Pi-model, and thus the effective capacitance.

In order to compute higher-order moments, an algorithm is required in which the computations start from the leaves. **Fig**. 7.6 shows a sub-tree with the source in B which is to be merged with a wire segment. The moments for the BC path, m_1^{BC} to m_i^{BC}, are already computed. The required moments can be then calculated as follows (see [3]):

$$m_i^{AB} = -\mathrm{Re}\{m_{i-1}^{AB} C_n + m_{i-1}^{AD} C_f\} \tag{7.1}$$

$$m_i^{AD} = m_i^{AB} - m_{i-1}^{AD} R_\pi C_f \tag{7.2}$$

where $m_0^{AB} = m_0^{AD} = 1$. The moments for the complete AC path can be afterwards determined by moment multiplication:

$$m_i^{AC} = \sum_{j=0}^{i} m_j^{AB} m_{i-j}^{BC}. \tag{7.3}$$

As shown in **Chap**. 4, there are several techniques that estimate the delay based on the first few moments. At this stage, there is no information available regarding physical hierarchy, interconnect topology, wire sizing, wire splitting, and therefore simple estimation delay models mostly based on worst-case methods have to be used. Because it is necessary to employ a delay estimator that is more accurate and yet not much more complex than the Elmore model, the D2M metric is used.

7.1.2 Simultaneous Placement and Buffer Planning

The objective of this section is to show that by considering repeater planning and coding early in the design flow, design engineers can achieve significant reduction of both dynamic and static power consumption in buffered interconnects. For this purpose, we have selected Simulated Annealing as placement algorithm mainly due to its easy-to-extend cost function. Thus, straightforward evaluations of different solutions can be performed.

Simply put, the fundamental idea of the employed method is to evaluate power metrics at every new solution generated during the placement procedure. For this purpose, the developed method computes for each new layout a total cost function consisting of the following weighted partial sub-costs: area, overlap, wirelength (half-perimeter), tree wirelength, buffer numbers, total interconnect capacitance, and required arrival time. Consequently, the method requires the construction of the buffered trees. Due to the fact that both buffer option computation and tree construction are performed in a bottom-up manner, they can be performed simultaneously which results in a time-efficient solution. In the sequel, we discuss the combination of the A-Tree algorithm with a modified van Ginneken algorithm and the eventual incorporation into placement.

The input of the A-Tree algorithm is a net together with a set of ports. The first port is always the source that determines the coordinate system. In order to achieve realistic results, the algorithm has been extended to all four quadrants of the coordinate system. The quadrants are treated separately and thus for each quadrant a sub-tree is obtained. When merging two sub-trees T_u and T_w, the nodes u and w and a potentially new Steiner node are passed to a method which immediately computes the buffer options. Because of the concomitant buffer option calculation and Steiner-tree construction, the algorithm had to be slightly modified.

As the goal is the minimization of power consumption in buffered interconnects, power must be estimated by means of metrics that can be computed during placement. On the one hand, the static component of power consumption depends on the supply voltage and the sum of the leakage currents. The total leakage current in buffered interconnects is proportional to the total area required by the planned repeaters. For reasons of restricting the computational complexity, identical buffers have been considered, which means that in the current scenario, the total number of buffers gives a good estimate of the total static power consumption. On the other hand, the dynamic component depends on the mean transition activity factor, the operating frequency, the square of the supply voltage, and the total switching capacitance. It is to be noticed that short-circuit currents are not considered, due to the fact that this component of dynamic power consumption is negligible in optimized interconnect structures [9, 52]. Considering that the supply voltage and the operating frequency are fixed and that the transition activity is given, in order to estimate the total dynamic power consumption in the buffered interconnects, the total switching capacitance must be approximated. This can be done by integrating the capacitance computation in the van Ginneken algorithm. Therefore, the total capacitance $c_{tot}(T_i)$ is carried together with the buffer options. Thus, an option Z_i becomes:

$$Z_i = \big(q(T_i), c(T_i), c_{tot}(T_i)\big). \tag{7.4}$$

When merging two options, the corresponding total capacitances are added and by adding a buffer, the total capacitance augments by the buffer capacitance, $c_{b_{tot}}$.

When determining the buffer positions in the case of the A-Tree algorithm, one has to differentiate between two cases. In contrast to **Fig**. 7.7 b), **Fig**. 7.7 a), shows two nodes

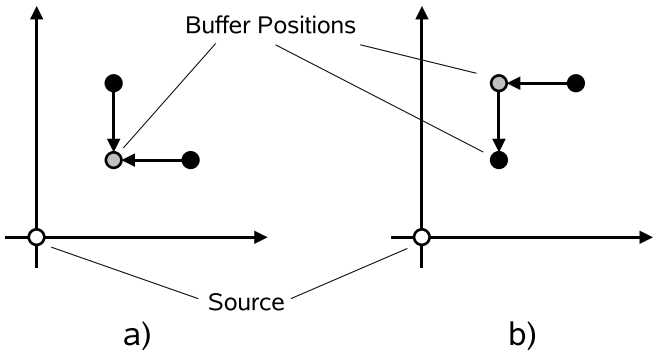


Fig. 7.7: Construction of a Steiner tree with buffer options

that are connected through an additional Steiner node. The Steiner node has thereby in the newly generated tree the minimal distance to the source. In the latter case, the three nodes are passed to the method *riseOptions(parent,child1,child2)*, where the Steiner node represents the parent. Afterwards, this method computes a buffer option for the parent node.

The calculation of the wire delay is also performed in a different manner. The applied algorithm cannot presuppose a completely constructed tree like in the original algorithm. Instead, the computation of interconnect delay from *child1* to *parent* and from *child2* to *parent* is performed at the very beginning of the *riseOptions()* method. Additionally, the resulted total capacitance has to be passed further. The moment-based delay computations previously presented start from the leaf nodes and the calculation of the Pi-model, moments, and C_{eff} can be therefore seamlessly integrated into the van Ginneken algorithm. The extended buffer insertion algorithm that has been integrated into placement is given in Listing 7.3.

In order to assess the effectiveness of the proposed method, several technology-related parameters had to be determined. For this purpose, a set of SPICE simulations with double-inverter buffers for a projected 65 nm technology node have been carried out. The employed transistor models have been taken from BSIM4[1] [40] and the interconnect resistances and capacitances for typical wire values in a 65 nm technology have been calculated with the formulas presented in **Chap**. 2. In order to achieve smaller buffer input capacitances and output resistances, the second (output) inverter has been designed to be three times larger than the first (input) one [6, 141]. The rise time has been set to t_r=100 ps. Moreover, several sets of 50 and 100 blocks with a total added layout area of 0.25 mm^2 have been considered. Following a bivariate Gaussian distribution, the area and the aspect ratio of each module have been varied between 400µ m^2 and 3100µ m^2, and from 0.4 to 1 respectively. Moreover, the number of nets per sink has also been varied. Thus, the influence of the net size can also be analyzed.

Simulations have shown that it is of utmost importance to include the HP estimation into the algorithm. Otherwise, the developed placement algorithm does not manage to

[1]Berkeley Short-channel IGFET (Insulated-Gate Field-Effect Transistor) Model

```
RISEOPTIONS(parent, child)
 1  /*Wire Delay Calculation*/
 2  Z_p ← GETOPTIONS(parent);
 3  Z_c ← GETOPTIONS(child);
 4  for z ∈ Z
 5  do π ← GETPI();
 6     M ← GETMOMENTS(child);
 7     for m ∈ M
 8     do m ← CALCMOMENT(π, m);
 9        π = CALCPI(e, π, );
10     Z_c ← Z_c ∪ (q_z^m in, c_z + c_e, c_tot + c_e, π, M);
11
12  /*Computation of Options without Buffer*/
13  i ← 1; j ← 1;
14  while i ≤ |Z_c| and j ≤ |Z_p|
15  do (q_l, c_l, c_tot_p, π_p, M_p) = Z_p;
16     (q_r, c_r, c_tot_c, π_c, M_c) = Z_c;
17     π ← CALCPI(π_p, π_c, );
18     Z ← Z ∪ {(min(q_p, q_c), c_l + c_r, c_tot_p + c_tot_c, π, M_p + M_c};
19     if q_l ≤ q_r
20        then i ← i + 1;
21
22     if q_r ≤ q_l
23        then j ← j + 1;
24
25  /*Insertion of Buffer Option*/
26  find (q, c, c_tot, π, M) ∈ Z
27       c_eff ← CALCCEFF(π)
28       q_b = q − D_buf(b, c_eff); is max
29  Z ← Z ∪ (q_b, c_b, c_tot + 4c_b, π_b, M_b);
30
31  parent ← SETOPTIONS(Z);
```

Listing 7.3: Buffer option calculation with accurate delay computation.

bring modules of the same net in a closed neighborhood. Further, when too many buffers, are inserted the total capacitance increases dramatically. Therefore, in order to achieve good results, the parameter weights of the cost function have to be chosen in a balanced manner. Actually, in order to choose suitable values for the weights, an initial plain area-oriented placement is performed.

The results are illustrated in the following tables. The rows show the obtained difference in area, half-perimeter, total tree length, total interconnect capacitance (including the input capacitances of the buffers), the required arrival time, and the number of inserted repeaters. As previously mentioned, the number of planned buffers and the total wire capacitance are direct measures of the static power consumption and the dynamic power consumption of the interconnect structure respectively. In order to check the con-

50 nets with 15 sinks per net						
	A [mm^2]	HP [mm]	Tree [mm]	C [pF]	RAT [ns]	Buffers
Area	0.2599	52.150	106.283	47.709	23.66	288
HP	101.3%	78.1%	80.4%	85.4%	100.1%	84.4%
Moment a)	107.7%	83.5%	84.4%	72.0%	100.6%	39.2%
Moment b)	101.3%	80.1%	82.9%	73.5%	100.6%	45.8%
100 nets with 5 sinks per net						
Area	0.2599	85.228	116.303	43.959	48.96	235
HP	101.9%	67.1%	69.4%	68.2%	99.9%	51.9%
Moment a)	100.1%	67.7%	71.0%	74.4%	101.2%	70.2%
Moment b)	107.7%	73.5%	75.0%	71.3%	101.0%	51.9%
50 nets with 1 to 50 sinks per net						
Area	0.2599	51.377	125.321	61.92	22.89	373
HP	100.1%	84.7%	90.0%	96.0%	100.6%	101.3%
Moment a)	110.3%	92.0%	94.9%	78.4%	99.3%	42.6%

Tab. 7.1: Simulation results for 100 normally distributed modules. Two different sets of optimization parameters have been considered for moment-based estimation

sistency of the achieved improvements, simulations with moment-based delay computations have been repeated for different weights. The results of the HP-method and the moments-based algorithm are reported with respect to the area-oriented one.

Tab. 7.1 shows that for the first type of nets the proposed method allows a buffer number reduction of more than 60 % compared to the plain area-minimizing method and of more than 50 % compared to the HP-Method. Moreover, the total interconnect capacitance has been reduced by 28 % and 18 % compared to the Area- and the HP-Method respectively. The price paid is an increase of less than 8 % in layout area. It is to be mentioned that when calculating the total area, the area of the inserted repeaters has not been taken into consideration. Therefore, because of the projected percentage of buffers in total chip area, we can actually expect a further reduction in total area. Additionally, the larger free inter-module spaces offer more degrees of freedom for final buffer placement and sizing as the gas stations (buffers are often grouped in blocks) are thus decongested and can be designed for smaller capacity. As expected, when dealing with small nets, even though in high number, the optimization possibilities are rather inexistent. In this case, the extended method is unable to improve neither the total capacitance, nor the required buffer number. In the third case, the number of sinks per net has been varied between 1 and 50. Employing the HP-Method yields no improvement in total capacitance and number of buffers. On the contrary, the proposed method allows a buffer reduction of 58 % and a saving in total capacitance close to 20 %. It is nonetheless no surprise that

25 nets with 10 sinks per net						
	A [mm^2]	HP [mm]	Tree [mm]	C [pF]	RAT [ns]	Buffers
Area	0.2524	21.804	39.530	17.23	11.98	101
HP	103.7%	96.5%	98.1%	96.7%	100.0%	93.1%
Moment a)	109.4%	84.0%	81.8%	72.2%	100.5%	42.6%
20 nets with 20 sinks per net						
Area	0.2524	18.963	47.223	22.43	9.36	135
HP	103.3%	86.1%	85.5%	74.3%	97.2%	43.0%
Moment a)	111.7%	93.9%	86.0%	70.4%	99.7%	31.1%
Moment b)	107.4%	91.7%	88.0%	71.6%	99.7%	31.9%

Tab. 7.2: Simulation results for 50 normally distributed modules. Two different sets of optimization parameters have been considered for moment-based estimation

sets of modules with many large nets offer much higher optimization opportunities than those with few small nets. This can be also seen in **Tab**. 7.2 for sets of 50 modules.

Furthermore, we can make two more observations. First, the reduction in buffer numbers is generally higher than the reduction in total capacitance. The rationale behind this is that the total capacitance of the interconnect structure is dominated in VDSM technology nodes by wire capacitance. Therefore, a significant reduction of the buffer number has a smaller impact on the total capacitance than reducing the total wire length. Secondly, a higher number of total nets offers more optimization possibilities than the total number of sinks. This can be explained by the fact that modules connected by a smaller amount of nets can be easier placed closely, and for this purpose, the half-perimeter is generally a good metric.

The simulations have been repeated for other sets of modules, namely: 100 inverse-normally distributed (**Tab**. 7.3), 100 almost identical (**Tab**. 7.4), and 100 randomly distributed modules (**Tab**. 7.5). The results are in accordance with the previous ones. Compared to the HP-method, the proposed algorithm yields a significant reduction in buffer

25 nets with 30 sinks per net						
	A [mm^2]	HP [mm]	Tree [mm]	C [pF]	RAT [ns]	Buffers
Area	0.2460	24.781	71.142	37.9	11.58	249
HP	105.0%	89.0%	90.6%	88.7%	100.3%	79.5%
Moment a)	113.0%	100.3%	90.9%	76.3%	99.5%	45.4%
Moment b)	115.7%	101.4%	92.1%	74.7%	99.1%	39.8%

Tab. 7.3: Simulation results for 100 inverse-normally distributed modules. Two different sets of optimization parameters have been considered for moment-based estimation

50 nets with 15 sinks per net						
	A [mm^2]	HP [mm]	Tree [mm]	C [pF]	RAT [ns]	Buffers
Area	0.2709	46.065	99.326	44.53	23.79	258
HP	101.3%	90.7%	89.7%	90.2%	100.0%	84.9%
Moment	101.3%	95.2%	95.4%	87.0%	99.8%	66.7%

Tab. 7.4: Simulation results for 100 uniformly distributed modules

number and total interconnect capacitance at the expense of very small increase in area, HP and total tree length. Moreover, the RAT is virtually unchanged.

Consequently, incorporating buffer planning during placement has been proven to allow significant improvements in terms of power consumption in interconnects. Nevertheless, as shown in **Tab**. 7.6, the main drawback of the presented algorithm lies in its complexity which explodes with increasing number of nets and sinks. By comparing the run-times, one can easily notice that the complexity increases mainly because of the buffer option calculation. By replacing the Elmore delay computation with the D2M-based one, the complexity of the method increases at a manageable rate. This shows that in order to implement a fast simultaneous placement and buffer planning, one has to focus on optimizing the estimation of required buffers for a given placement.

The proposed method shows that generally the total wire capacitance, and thus the dynamic power consumption in the interconnect structure, can be reduced by around 20%. Nonetheless, the method is not universally effective. If a layout is characterized by rather small nets, although numerous, one has to expect less significant improvements, if any. On the contrary, huge optimization possibilities have been observed when dealing with large nets. Further, the simultaneous placement and buffer insertion achieves an even more significant reduction in total area required for buffer planning. As leakage-induced power consumption is increasing dramatically with every new technological node and because of the steadily augmenting number of buffers in large high-density VDSM designs, the developed method implies a substantial decrease in buffer area and thus in buffer-induced static power consumption. Moreover, due to the fact that actually the buffers themselves require an increasing portion of the total die area, incorporating

50 nets with 15 sinks per net						
	A [mm^2]	HP [mm]	Tree [mm]	C [pF]	RAT [ns]	Buffers
Area	0.2625	44.225	95.818	44.73	23.71	275
HP	92.4%	76.9%	77.3%	84.2%	101.3%	85.1%
Moment	101.3%	97.5%	92.0%	83.7%	100.4%	64.0%

Tab. 7.5: Simulation results for 100 quasi-identical modules

	50 nets (15 sinks per net)	100 nets (5 sinks per net)
HP	33 min	36 min
Elmore	390 min	107 min
Moment	708 min	152 min

Tab. 7.6: Comparison of run-times

buffer planning early in the design flow, i.e during placement and/or initial floorplanning, can potentially mean a worthy of mention reduction in chip area [56, 165].

Signal encoding can be integrated in the simultaneous placement and buffer insertion in order to further improve performance and/or power consumption. On the one hand, we can improve performance by not permitting selected input patterns, and on the other hand, we can reduce the dynamic power consumption while maintaining the same performance. In the latter case, the worst case switching pattern are prohibited and the bus operating frequency can be slowed down until the resulting troughput is equal to that of the uncoded bus.

Tab. 7.7 illustrates coding-based improvements in terms of power consumption for some simplified scenarios. After the simultaneous placement and buffer insertion, we have assigned to the resulting interconnects random physical width and bus width for a fixed wire sheet geometry. That following, the methodology for calculating the coding-based throughput improvement described in **Chap**. 6 has been applied. The resulting throughput improvement has been employed to compute the maximum frequency slowdown for maintaining the initially obtained throughput. This operating frequency reduction has been then used for approximating the decrease in power consumption with respect to the uncoded case. The maximum allowed delay has been chosen equal for all interconnects (Δ_4 and Δ_3), as well as arbitrarily for each interconnect (the two arbitrary cases a) and b) indicated in **Tab**. 7.7). It is to be mentioned that the computed values are approximative as the cost of coding itself is neglected. Nonetheless, the encoding is expected to be even more efficient if integrated in the simultaneous placement and buffer insertion algorithm.

	50 nets (15 sinks per net)	50 nets (1 to 15 sinks per net)	100 nets (5 sinks per net)
Δ_4	93.3 %	89.3 %	90.7 %
Δ_3	86.4 %	87.1 %	84.7 %
Arbitrary Δ-s a)	82.1 %	78.6 %	83.9 %
Arbitrary Δ-s b)	79.1 %	80.2 %	80.7 %

Tab. 7.7: Reducing power through signal encoding

7.2 Interconnect-Centric Design Flow Integration

Very deep sub-micron effects, and interconnect structures in particular, are considered to be a potential showstopper to the continuation of Moore's law [191]. There is however a strong relation not only to axiomatic physical limits but also to fundamental limitations of the current CAD methodologies and targeted architectures. In order to efficiently improve the latter two, there is a stringent need to stop treating interconnects as an afterthought and to place them inside the core of the design flow. As a result of such a paradigm shift, a new set of interconnect-related algorithms, methodologies, design and estimation metrics have to be integrated into the design flow. Moreover, with interconnects translated at the center of the design flow and design metrics changed, problem definitions are also to be reformulated.

Fig. 7.8 illustrates an interconnect-centric design flow that also binds signal encoding. It consists of five important design phases that strongly interact with each other: system-level specification, interconnect planning, interconnect synthesis, interconnect refinement, and interconnect layout. The last design step is beyond the scope of this thesis and is therefore not mentioned in the following. A fundamental characteristic is that not only do the estimation models employed at every abstraction level have to be enhanced and ameliorated by interacting with lower levels, but the used design metrics also require to be iteratively revised.

7.2.1 Design and Architecture Specification

A severe limitation of the majority of design flows is the level of abstraction at which design engineers must enter. In order to address in an effective manner the design of complex, heterogeneous systems, design description must start at higher levels of abstraction. Capturing the design at high levels does not just allow exploiting the vastly available degrees of freedom, but it also represents the only way to reduce design costs and design time, or at least keep them manageable. Therefore, flexibility is an essential aspect and platform-based design emerged as a promising architecture paradigm [80, 163].

Reusing blocks like versatile processors, high-performance dedicated circuits, or reconfigurable logic modules, provides several advantages: shorter time-to-market, extended product life-cycle and functionality on demand. Moreover, IP (Intellectual Property) reuse allows designers to skip important parts of the tedious hardware verification process. The configuration for the reconfigurable part or the software for the programmable part can be developed in parallel to the platform design, offering thus great flexibility to late design changes. Thus, functionality can be changed after system deployment or even during operation.

Probably the most groundbraking change in reformulating the design problem at higher levels of abstraction is related to defining the design goals and choosing the appropriate design metrics, which are deeply related with fundamental challenges and hitches

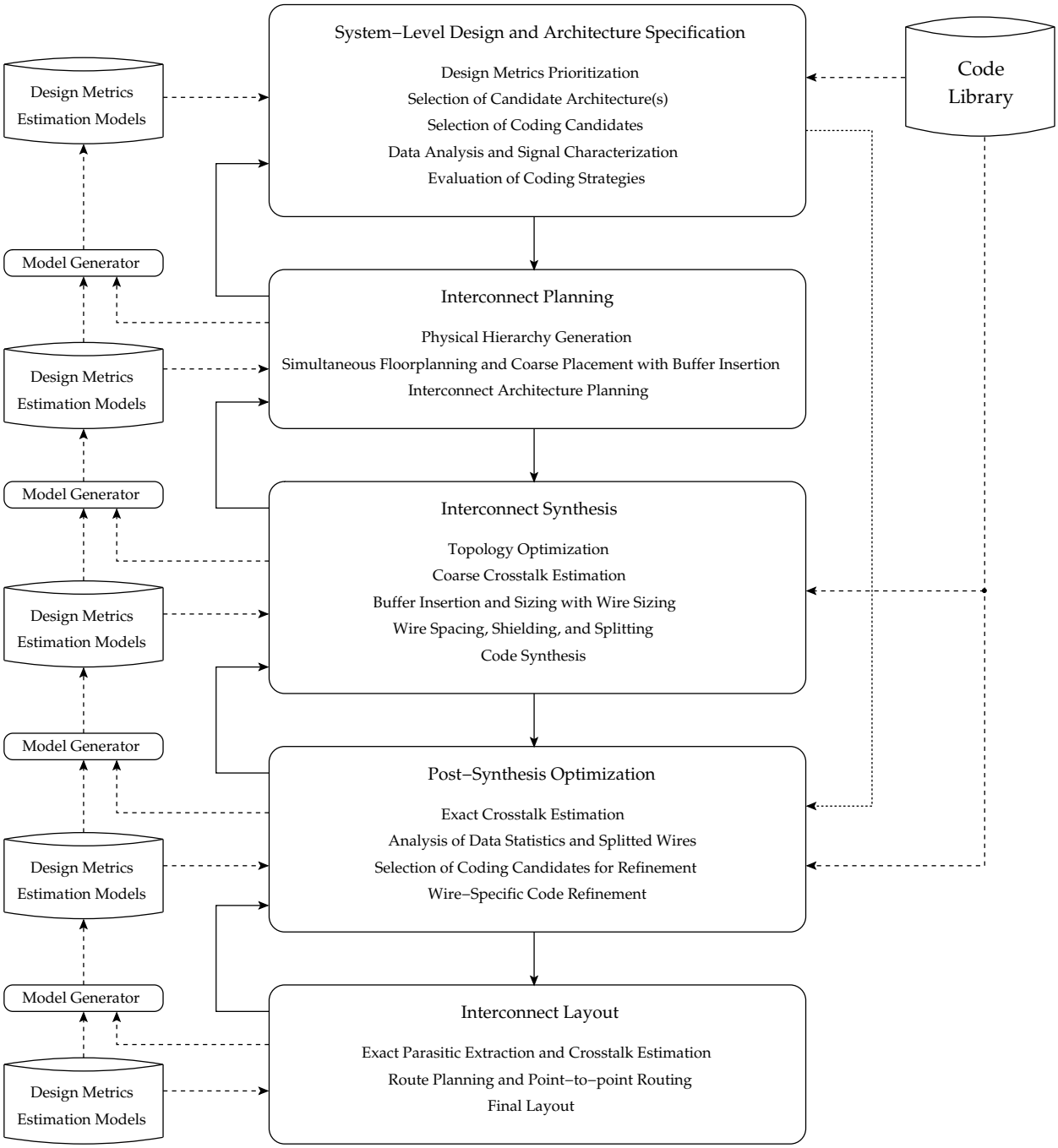


Fig. 7.8: Proposed interconnect-centric design flow and signal encoding binding

at lower levels. Moreover, those design metrics have to be reiterated during the design process itself as specifications and design requirement are often incomplete at design start and mostly evolve over time [24]. Therefore, incremental design combined with revalidation steps are of an increasing importance.

Thus, the selection of one candidate architecture or another – be it an application-specific architectural template, a general purpose microarchitecture, or a dedicated fixed design – represents an iterative process too. The design input required at system-level

in order to efficiently choose the target architecture is a statistical and/or simulative description of the operating data. Furthermore, it is of paramount importance to prioritize the chosen design metrics like power consumption, performance, or flexibility, also in a recurrent manner. In this way, an application-specific trade-off can be defined and finally optimized.

For instance, in the case of DSP architectures, concept engineers must deliver information about functional blocks and the data throughput required to be supported by the communication architecture for real-time operation. On this basis, estimations on the transmitted data and on the number and dimensions of the required buses or point-to-point interconnects are done. As a result, different coding schemes that improve performance and power can be evaluated. On the one hand, it is important to find out how the specific data characteristics (f.i. data correlation) and communication requirements can be exploited. On the other hand, it is decided whether or not to use redundancy in order to increase throughput and/or reduce power consumption. Furthermore, bit-level, word-level transition activities (or just the breakpoints – depending on the requirements and the design flow) are stored in a data base for later use during the synthesis and refinement steps.

7.2.2 Interconnect Planning and Synthesis

Interconnect Planning and Interconnect Synthesis are two strongly interrelated design steps that interact substantially before the interconnect layout can be performed. Interconnect planning is the first step after design specification and it has a tremendous impact on the subsequent design phases. Therefore, the existing degrees of freedom have to be exploited in accordance with the synthesis step and its design metrics.

Interconnect Planning can be divided into three other sub-steps: physical hierarchy generation, communication planning (synchronous, multi-cycle or asynchronous), simultaneous floorplanning and coarse placement with buffer insertion, and interconnect architecture planning. The physical hierarchy defines de facto the global, intermediate, and local interconnects. The goal of floorplanning and coarse placement is to select a good topology and approximate wire width, spacing, and layer assignment [32]. It has to be performed at the same time with buffer insertion. The objective of interconnect architecture planning is to identify overall interconnect parameters like number of layers and layer geometry.

In order to keep the right balance between required accuracy and design complexity, first or second order estimators have to be employed along with more accurate and complex design metrics. For instance, if crosstalk cannot be accurately estimated at a certain design stage, one can work first with worst case approaches for noise and delay. For the latter, a combination between the more simple Elmore and D2M (or other simple delay metric based on the first few moments) and the more accurate ELD model can be envisaged. Moreover, general rules have to be respected in order to permit an optimal

interconnect synthesis afterwards. In this context, buffer insertion must be optimized not only for power and performance but also for crosstalk. For instance, a general rule of thumb is to insert more often smaller buffers rather than a few large ones. Thus, crosstalk is reduced, as the set of capacitive aggressors is reduced and long current return paths are avoided. Furthermore, the so-called buffer gas stations must also be planned.

Interconnect synthesis is probably the most complex design step as its goal is to solve a wide span of optimization problems: appropriate topology construction; sizing and positioning of buffers; coarse crosstalk estimation; ordering, shaping, sizing, spacing, shielding, and splitting of wires. The coarse crosstalk estimation can be performed only after a rough topology is constructed and basic buffer planning is realized. Buffers are located in their final positions and after wire shaping, shielding, and splitting, and based on the analysis of data characteristics performed at system-level, the selected codes are synthesized. Those codes are inserted afterwards in the communication structures or into the communicating blocks or modules, depending on the design flow and final architecture. It is to be mentioned that the synthesis of any code is highly dependent not only on data and interconnect geometry, but also on interconnect architecture: shared or dedicated, synchronous or asynchronous, globally synchronous locally asynchronous (GALS) or multi-cycle.

Interconnect refinement is the step in the design flow where coding for performance, power, and/or crosstalk is refined. This design step can be regarded as a part of the synthesis step. However, there are some conceptual differences. In general, code synthesis and physical wire optimization techniques are methods that are associated with different levels of abstraction. On the one hand, signal encoding refers to adding redundancy in the transmitted data at higher levels of abstraction, while on the other hand, wire spacing, sizing, and splitting are techniques that optimize crosstalk, delay, or power at lower levels of abstraction. The main objective of interconnect refinement is to bring together signal encoding and physical wire optimization techniques, as the synthesized codes can be refined in order to take advantage also of those physical wire optimization methods. Therefore, the refinement is performed in a wire-specific way.

Even though FPGAs dispose of vast interconnection structures that are mainly responsible for the high flexibility, an essential characteristics of FPGAs is that an important percentage of the available logic capacity generally remains unused due to interconnect limitations [21]. This characteristic is in general typical also for application-specific platforms, though at a lower magnitude. Schemes implementing coding for throughput, power, and crosstalk require some additional localized logic around the interfaces between large interconnect structures and computational units. As this logic comes for free in reconfigurable logic and architectural templates, coding schemes can be synthesized and afterwards optimized in a post-place-and-route step. Obviously, information about the transmitted data is required together with an application-specific code library as previously mentioned.

7.3 Summary

With technology improvements, two main trends can be identified. On the one hand, the steadily increase in power dissipation and density poses severe difficulties on thermal management, reliability, power distribution, and efficient low-cost packaging. On the other hand, both power consumption and performance limitations are more and more decisively influenced by the interconnect structure. Therefore, power and performance in interconnect structures are design metrics that have to be taken into account early in the design-flow.

The contribution of this chapter is twofold. It has been shown on the one hand, that an interconnect- and buffer-centric placement is able to take full advantage of the high optimization opportunities typical for the early design stages. By extending an SA-based placement technique with efficient Steiner-tree construction, moment-based delay calculation, and van Ginneken's buffer insertion algorithm, significant reductions in total interconnect structure capacitance and total repeater area have been reported especially when appending coding for throughput improvement. It has been shown that both performance and power can be improved by combining buffer planning with signal encoding schemes.

On the other hand, an interconnect-centric design flow that also binds signal encoding schemes has been introduced. Signal encoding schemes are first evaluated in conjunction with signal characterization and interconnect architecture selection during the system-level design and architecture specification step. Afterwards, interconnect planning is responsible for physical hierarchy generation, simultaneous floorplanning and buffer insertion, as well as for the general interconnect architecture planning. During interconnect synthesis, signal encoding schemes are synthesized, based on the results of the initial high-level data analysis and coding evaluation phase. Moreover, buffers are sized and placed at their definitive locations, while the wire geometry is also finalized (sizing, spacing, shielding, splitting). These physical wire optimization techniques are merged with coding improvement during a so-called interconnect refinement design step.

Chapter 8

Concluding Remarks

Contents

This thesis introduced a signal-encoding-based methodology for improving power consumption and performance in very deep sub-micron interconnect structures. By employing a pattern-dependent delay model and a statistical power macromodel, several encoding schemes for improvement of power consumption and performance are constructed and analyzed. Thus, a methodology to construct and analyze different codes has been developed. The built codes and the proposed methodology can be integrated in an interconnect-centric design flow that exploits the optimization opportunities available especially at high-levels of abstraction as well as those resulting for refinement during and after interconnect synthesis.

8.1 Contributions of the Work

In **Chap**. 2, the dependency of crosstalk, delay, and power consumption in very deep sub-micron interconnects on the input patterns has been proved. It has been shown that capacitive and inductive coupling have antagonistic effects on delay and that inductive coupling does not influence the switching component of dynamic power consumption. Nevertheless, inductive effects modify rise and fall times, affecting therefore the short-circuit component of the dynamic power consumption. The short-circuit power consumption represents usually a very small fraction in optimized interconnects that can be therefore neglected.

Upon this basis, a pattern-dependent delay macromodel – the so-called extended delay model (ELD) – has been constructed in **Chap**. 4. The model accurately predicts the

delay in both capacitively and inductively coupled lines (**Eq**. (4.22) and **Eq**. (4.30)). Moreover, the model has been extended to incorporate also the effects of process variations (**Eq**. (4.32)). Further, a power macromodel is constructed (**Eq**. (4.35)–**Eq**. (4.42)) that can be used also in the case of non-symmetrical buses (**Eq**. (4.45)–**Eq**. (4.49)). The previously mentioned models can be employed to assess the effectiveness of coding for performance and power consumption.

Chap. 5 dealt first with the analysis of the bit-level transition activity in typical DSP signals. Based on the characteristics of self activity and coupling activity, several hybrid codes have been constructed. Those codes are the result of combining redundant codes with non-redundant ones, in order to efficiently exploit the spatial and temporal correlation in typical DSP signals. The effectivity of the schemes has been assessed by means of extensive simulations on both synthetic (**Fig**. 5.8–**Fig**. 5.17) and real data (**Tab**. 5.1 and **Tab**. 5.2). Another essential contribution of the chapter is the outcome that partial bus invert schemes can be easily constructed by determining the so-called MSB and LSB breakpoints. In this context, two partial bus invert schemes have been constructed – PBIH and PBIC – for reducing self activity and coupling activity, respectively, as well as a partial odd/even bus invert (POEBI) architecture (see **Fig**. 5.18, **Fig**. 5.19, **Tab**. 5.3 and **Tab**. 5.4). The developed POEBI scheme significantly reduces the coupling transition activity. Moreover, the PBI and POEBI codes were modified to construct adaptive versions of the partial bus invert schemes (APBI and APOEBI – **Fig**. 5.21 and **Fig**. 5.22). The adaptive schemes are extremely versatile as they have the ability to adjust to varying data characteristics. Further, limits for self and coupling (total) transition activities have been derived (**Fig**. 5.24–**Fig**. 5.28).

The main goal of **Chap**. 6 was to determine and analyze fundamental limits of coding for performance. First, benefits and drawbacks of state and transition coding have been highlighted. The pattern-dependent delay model has been employed to show how delay classes identified in capacitively coupled buses mutate to dissoluted values that define rather intervals with increasing inductive effects (**Tab**. 6.1). Afterwards, exact limits for state coding and bounds for limits in the case of transition coding have been computed (**Eq**. (6.26), **Eq**. (6.36), **Eq**. (6.56), and **Fig**. 6.3). Moreover, limits for the bus aspect factor have been calculated in order for the coding schemes to be efficient (**Ineq**. (6.27) and **Ineq**. (6.37)). Thereafter, simple codings for throughput have been constructed and evaluated, namely the D-RLL(1,∞) and M-RLL(2,∞) schemes. Furthermore, spacing and shielding have been analyzed and compared by employing the same pattern-dependent delay model. That model can be easily used to choose among the best anti-crosstalk and delay improvement techniques as shown also in **Sec**. 6.3.3. The issue of simultaneously addressing delay improvement and transition activity reduction through coding has been treated in **Sec**. 6.4. The mean delay has been linked to the average dynamic power consumption and it has been highlighted that reducing the mean delay and decreasing the transition activity are two related problems. However, reducing the mean delay does not say anything about improving the line delay, because line delay is solely defined by the

worst-case no matter how often that appears. It has been shown that in the case of differential coding schemes like D-RLL(1,∞), self activity can be easily reduced by assigning the codewords with minimum weight to those with the highest probability of appearance (**Tab**. 6.6). Lastly, the generalized power macromodel has been employed in order to construct a spacing-based design-time encoding and an active-shielding-like run-time one for optimizing delay and total transition activity. Thus, the effectiveness of combining coding with lower level techniques like spacing and shielding is demonstrated. This represents an essential contribution of this work.

Finally, **Chap**. 7 introduces an interconnect-centric design flow in which coding can be seamlessly integrated. Coding techniques can be first analyzed at higher levels of abstraction (system-level design and architecture specification), synthesized (interconnect synthesis) and eventually optimized in a wire specific manner (interconnect refinement). Moreover, in order to prove the enormous optimization opportunities available at high levels of abstraction during the early stages of the design, a simultaneous placement and buffer planning algorithm employing simple delay metrics has been constructed. The algorithm achieves a significant reduction in power consumption, total buffer area, and routing resources while keeping performance in the same range as other placement methods. In addition, coding for throughput has been appended to the developed buffer planning. It has been shown that in this way, that performance and/or power consumption can be further improved.

8.2 Directions for Future Work

The proposed interconnect-centric design flow represents a solid backbone for a design flow that tries to cope with challenging VDSM effects. However, coding represents only one effective method to reduce crosstalk, line delay, and improve power consumption. Several directions for future work can be envisaged.

Development of further codes that simultaneously improve performance and power consumption: The described methodology can be employed in order to derive more generic or application-specific hybrid encoding schemes that optimize/improve also other figures of merit, f.i. the power-delay product.

Extension of the developed delay and power macromodels: The developed macromodels can be extended to include correlated random and process variations. Furthermore, the delay power macromodel can be enhanced to be interfaced with more accurate gate models. Variations of the macromodels can be envisaged in order to meet different complexity-computability trade-offs for the specific needs of a design flow or another.

Advanced Buffer Insertion Methods: Signal encoding schemes can also be assessed during buffer planning in a more seamlessly integrated manner. However, in order to be able to do so, buffer planning algorithms have to be extended with basic architecture planning, topology construction, and physical hierarchy generating methods. Moreover, wire spacing algorithms – at least first-order ones – must be incorporated into repeater planning. Such a buffer planning is in fact equivalent to a very complex interconnect-centric design flow characterized by an even stronger interaction between interconnect planning, synthesis, and refinement.

Combining several power and performance optimization techniques and integration into an interconnect-oriented CAD framework: The most evident subsequent step is the integration of various techniques like accurate delay and transition activity estimation, buffer and gas station planning, signal encoding, wire sizing, shaping, spacing, splitting, and shielding into the same interconnect-centric design flow in order to assess their effectivity on complex design problems. Further, code synthesis could be integrated into high-level communication synthesis methodologies.

Appendix A

The Trigonometric Solution of the Cubic Equation

The so-called complete cubic equation:

$$ax^3 + bx^2 + cx + d = 0,$$

with $a, b, c, d \in \mathbf{R}$ and $a \neq 0$, can be reduced by means of the substitution $x = y - \frac{b}{3a}$ to an incomplete cubic equation, i.e. its so-called canonical form:

$$y^3 + py + q = 0,$$

where the coefficients p and q are defined as:

$$\begin{aligned} p &= \frac{c}{a} - \frac{b^2}{3a^2}, \\ q &= \frac{d}{a} - \frac{bc}{3a^2} + \frac{2b^3}{27a^3}. \end{aligned}$$

Cardano's method. Let D be the discriminant of the incomplete cubic equation:

$$D = \left(\frac{p}{3}\right)^3 + \left(\frac{q}{2}\right)^2.$$

By defining the following coefficients:

$$P, Q = \sqrt[3]{-\frac{q}{2} \pm \sqrt{D}},$$

the solutions of the incomplete cubic equation are:

$$\begin{aligned} y_1 &= P + Q, \\ y_{2,3} &= -\frac{P+Q}{2} \pm i\frac{P-Q}{2}\sqrt{3}. \end{aligned}$$

Trigonometric solution. If $p, q \in \mathbf{R}$, then the roots of the incomplete cubic equation can be expressed elegantly with trigonometric functions as shown below [149].

Case 1: $D < 0$ $(p < 0)$. The solutions of the canonical equation are:

$$\begin{aligned} y_1 &= 2\sqrt{-\frac{p}{3}} \cdot \cos\frac{\alpha}{3}, \\ y_{2,3} &= -2\sqrt{-\frac{p}{3}} \cdot \cos\left(\frac{\pi}{3} \pm \frac{\alpha}{3}\right), \end{aligned}$$

where α is defined as:

$$\cos\alpha = -\frac{q}{2\sqrt{-\left(\frac{p}{3}\right)^3}}.$$

Case 2: $D \geq 0$ $(p < 0)$. The solutions of the canonical equation are:

$$\begin{aligned} y_1 &= -2\sqrt{-\frac{p}{3}} \cdot \frac{1}{\sin 2\varphi}, \\ y_{2,3} &= \sqrt{-\frac{p}{3}} \cdot \frac{1 \pm i\sqrt{3}\cos 2\varphi}{\sin 2\varphi}, \end{aligned}$$

where:

$$\begin{aligned} \tan\varphi &= \sqrt[3]{\tan\frac{\theta}{2}}, \quad |\varphi| \leq \frac{\pi}{4}, \\ \sin\theta &= \frac{2}{q}\sqrt{-\left(\frac{p}{3}\right)^3}, \quad |\theta| \leq \frac{\pi}{2}. \end{aligned}$$

Case 3: $D > 0$ $(p > 0)$. The solutions of the canonical equation are:

$$\begin{aligned} y_1 &= -2\sqrt{\frac{p}{3}} \cdot \cos 2\varphi, \\ y_{2,3} &= \sqrt{\frac{p}{3}} \cdot \frac{\cos 2\varphi \pm i\sqrt{3}}{\sin 2\varphi}, \end{aligned}$$

where:

$$\begin{aligned} \tan\varphi &= \sqrt[3]{\tan\frac{\theta}{2}}, \quad |\varphi| \leq \frac{\pi}{4}, \\ \tan\theta &= \frac{2}{q}\sqrt{\left(\frac{p}{3}\right)^3}, \quad |\theta| \leq \frac{\pi}{2}. \end{aligned}$$

Consequently, the solutions of the complete cubic equation are:

$$x_k = y_k - \frac{b}{3a}, \quad \text{where} \quad k = \{1, 2, 3\}.$$

Appendix B

Markov Chains

A Markov chain is a sequence of random variables $\{X_i\}$, with $i \geq 0$, with the so-called Markov property, namely that, given the present state, the future and past states are independent. Formally,

$$\Pr(X_{n+1} = x | X_n = x_n, \ldots, X_1 = x_1, X_0 = x_0) = \Pr(X_{n+1} = x | X_n = x_n).$$

The possible values of X_i form a countable set S called the state space of the chain. Discrete Markov chains are often described by a directed graph, where the edges are labeled by the probabilities of going from one state to the other states. An example is given in **Fig**. B.1. Informally, the Markov property says that given the past history of the process, future behavior only depends on the current value. Markov chains are also known as first-order or lag-one Markov processes [121, 136].

A left stochastic matrix is a square matrix whose columns are probability vectors, i.e. the entries in each column are nonnegative real numbers whose sum is 1. Likewise, a right stochastic matrix is a square matrix whose rows are probability vectors. In a doubly stochastic matrix, all rows and all columns are probability vectors. Stochastic matrices can be considered representations of the transition probabilities of a finite Markov chain.

The probability of going from state i to state j in n time steps is defined as:

$$p_{ij}^{(n)} = \Pr(X_n = j \mid X_0 = i)$$

and the single-step transition is:

$$p_{ij} = \Pr(X_1 = j \mid X_0 = i)$$

The n-step transition satisfies the Chapman-Kolmogorov equation:

$$p_{ij}^{(n)} = \sum_{r \in S} p_{ir}^{(k)} p_{rj}^{(n-k)},$$

for any $0 < k < n$. The marginal distribution $\Pr(X_n = x)$ is the distribution over states at time n. The initial distribution is $Pr(X_0 = x)$. The evolution of the process through one

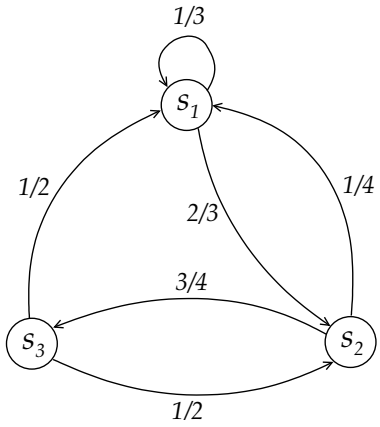


Fig. B.1: Example of a Markov process represented by a weighted directed graph (after [121])

time step is described by:

$$\Pr(X_{n+1} = j) = \sum_{r \in S} p_{rj} \Pr(X_n = r) = \sum_{r \in S} p_{rj}^{(n)} \Pr(X_0 = r).$$

If P is an $m \times m$ left stochastic matrix, then a steady-state vector or equilibrium vector for P is a probability vector w such that:

$$P \cdot w = w \quad \text{or} \quad w^t \cdot P^t = w^t.$$

The Stochastic Matrix Theorem (STM) says that if P is a regular stochastic matrix, then P has a steady-state vector w so that if w_0 is any initial state and $w_{n+1} = Pw_n$ for $n \geq 0$, then the Markov chain $\{w_n\}$ converges to w as n goes to infinity. That is:

$$\lim_{n \to \infty} P^n \cdot w_0 = w.$$

The above equation can also we written as:

$$\lim_{n \to \infty} P^n = [\underbrace{w \; w \ldots w}_{m}],$$

where $\sum_{j=1}^{m} w_j = 1$, and w_j and $w = [w_1, w_2, \ldots, w_m]$ represent the so-called asymptotic probability for state j and the state (or equilibrium) distribution vector, respectively.

The probability matrix of the Markov process represented in **Fig**. B.1 is:

$$P = \begin{pmatrix} \frac{1}{3} & \frac{2}{3} & 0 \\ \frac{1}{4} & 0 & \frac{3}{4} \\ \frac{1}{2} & \frac{1}{2} & 0 \end{pmatrix}.$$

Thus, the fixed vector of P is:

$$w = [0.34884 \quad 0.37209 \quad 0.27907]^t.$$

Appendix C

Capacity of Discrete Noiseless Constrained Channels

For a discrete constrained channel, a set S of all N permitted states can be defined as:

$$S = \{S_0, S_1, \ldots, S_{N-1}\},$$

where each state corresponds to one or more symbols that appeared in the current input sequence. The succession of channel states can be represented in two ways: state diagrams or trellis diagrams.

Consider the case of a binary RLL(0,3) channel (see **Fig**. C.1 for the state diagram). Thus, any sequence containing more than 3 consecutive symbols of the same type is interdicted. Such a channel admits 6 possible states:

$$S_0 = 111,\ S_1 = 11,\ S_2 = 1,\ S_3 = 0,\ S_4 = 00,\ S_5 = 00.$$

The state transition can be also illustrated by means of the trellis diagram given in **Fig**. C.2. It can be observed that a transition from one state to another is realizable in a certain clock cycles.

The discrete constrained channel can also be described by means of the state transition matrix, B. The entries b_{ij} of B represent the number of edges ending in S_j that start in S_i.

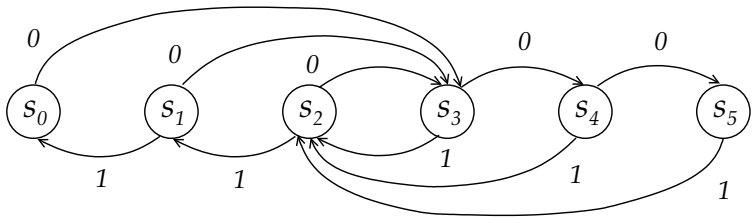


Fig. C.1: State diagram of an RLL(0,3) channel (after [121])

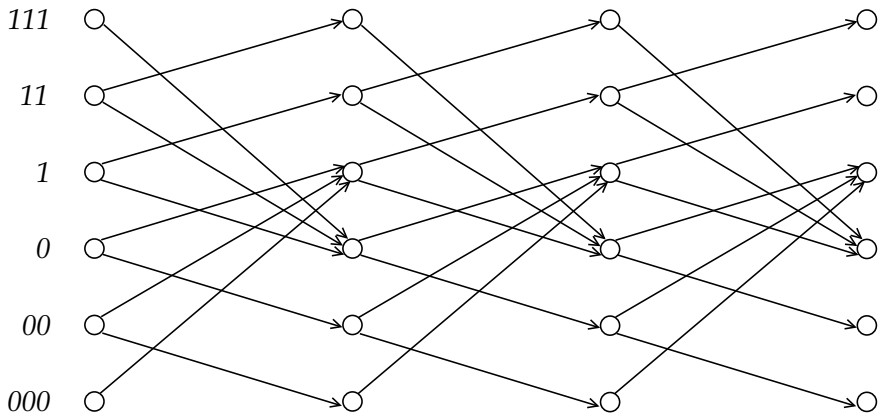


Fig. C.2: Trellis diagram for an RLL(0,3) channel (after [121])

Thus,

$$B = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}.$$

The description can be extended to higher number of steps. It can be shown that the entries $b_{ij}^{(n)}$ of B^n give the number of distinct edges of length n from state S_i to state S_j [121].

The state transition matrix B – or simply transition matrix – specifies only the allowed paths between states without taking into account the duration of the transmitted symbols. In order to cope with this problem, the extended state transition matrix, $B(x)$, is defined. The entries of $B(x)$, $b_{ij}(x)$, are defined as:

$$b_{ij}(x) = \sum_k x^{-\tau_{ij,k}},$$

where $\tau_{ij,k}$ represents the duration of the input symbol on position k, that can appear in state S_i and determines a transition in state S_j. The transition matrix B can be found for $x = 1$. In the previously considered example, the symbols have the same duration and thus:

$$B(x) = \begin{pmatrix} 0 & 0 & 0 & x^{-1} & 0 & 0 \\ x^{-1} & 0 & 0 & x^{-1} & 0 & 0 \\ 0 & x^{-1} & 0 & x^{-1} & 0 & 0 \\ 0 & 0 & x^{-1} & 0 & x^{-1} & 0 \\ 0 & 0 & x^{-1} & 0 & 0 & x^{-1} \\ 0 & 0 & x^{-1} & 0 & 0 & 0 \end{pmatrix}.$$

Similarly, the second-order extended state transition matrix is equal to $B(x)^2$, and $B(x)^n$ is the n-th order extended state transition matrix.

The capacity, ϱ, of a discrete noiseless constrained channel is defined as:

$$\varrho = \lim_{T\to\infty} \frac{\log_2 M(T)}{T} \text{ [bits/second]},$$

where $M(T)$ represents the number of permitted sequences that can be formed by the channel alphabet in a time T. If the symbols are of equal durations, the capacity can be expressed also as:

$$\varrho = \lim_{n\to\infty} \frac{\log_2 M(n)}{n} \text{ [bits/symbol]},$$

where $M(n)$ represents the number of sequences of length n. Further, it can be shown that the capacity of a discrete noiseless constrained channel with equal symbol durations can be calculated as:

$$\varrho = \log_2 \rho(B) = \log_2 \lambda_{max},$$

where $\rho(B)$ represents the spectrum of matrix B, i.e. the maximum (positive) eigenvalue of B. In the abovementioned case, $\lambda_{max} = 1.84$, and thus $\varrho = 0.88$ [bits/symbol].

References

[1] R. ACHAR and M. S. NAKHLA. Simulation of High-Speed Interconnects. *Proceedings of the IEEE*, 89(5):693–728, May 2001.

[2] K. AGARWAL, D. SYLVESTER, and D. BLAAUW. An Effective Capacitance Based Driver Output Model for On-Chip RLC Interconnects. In *Design Automation Conf. (DAC)*, pages 376–381, Anaheim, California, June 2003.

[3] C. J. ALPERT, A. DEVGAN, and S. T. QUAY. Buffer Insertion With Accurate Gate and Interconnect Delay Computation. In *Design Automation Conf. (DAC)*, pages 479–484, New Orleans, Louisiana, June 1999.

[4] R. ARUNACHALAM, F. DARTU, and L. T. PILEGGI. CMOS Gate Delay Models for General RLC Loading. In *IEEE Intl. Conf. on Computer Design (ICCD)*, pages 224–229, Austin, Texas, Oct. 1997.

[5] S. AXLER. *Linear Algebra Done Right*. Springer, New York City, 2nd edition, 1997.

[6] H. B. BAKOGLU. *Circuits, Interconnections, and Packaging for VLSI*. Addison Wesley, Reading, Massachusetts, 1990.

[7] M. W. BEATTIE and L. T. PILEGGI. Inductance 101: Modeling and Extraction. In *Design Automation Conf. (DAC)*, pages 323–328, Las Vegas, Nevada, June 2001.

[8] A. BELLAOUAR and M. I. ELMASRY. *Low-Power Digital VLSI Design. Circuits and Systems*. Kluwer, Norwell, Massachusetts, 1995.

[9] L. BENINI and G. DE MICHELI. *Dynamic Power Management. Design Techniques and CAD Tools*. Kluwer, Norwell, Massachusetts, 1998.

[10] L. BENINI and G. DE MICHELI. Networks on Chips: A New SoC Paradigm. *IEEE Computer*, 35(1):70–78, Jan. 2002.

[11] L. BENINI, G. DE MICHELI, A. MACII, E. MACII, and M. PONCINO. Reducing Power Consumption of Dedicated Processors Through Instruction Set Encoding. In *Great Lakes Symp. on VLSI (GLSVLSI)*, pages 8–12, Lafayette, Louisiana, Feb. 1998.

[12] L. BENINI, G. DE MICHELI, E. MACII, M. PONCINO, and S. QUER. System-Level Power Optimization of Special Purpose Applications: The Beach Solution. In *Intl. Symp. on Low Power Electronics and Design (ISLPED)*, pages 24–29, Monterey, California, Aug. 1997.

[13] L. BENINI, G. DE MICHELI, E. MACII, D. SCIUTO, and C. SILVANO. Asymptotic Zero-Transition Activity Encoding for Address Buses in Low-Power Microprocessor-Based Systems. In *Great Lakes Symp. on VLSI (GLSVLSI)*, pages 77–82, Urbana-Champaign, Illinois, Mar. 1997.

[14] L. BENINI, G. DE MICHELI, E. MACII, D. SCIUTO, and C. SILVANO. Address Bus Encoding Techniques for System-Level Power Optimization. In *Design Automation and Test in Europe (DATE)*, pages 861–866, Paris, France, Feb. 1998.

[15] L. BENINI, A. MACII, E. MACII, M. PONCINO, and R. SCARSI. Architectures and Synthesis Algorithms for Power-Efficient Bus Interfaces. *IEEE Trans. on Computer-Aided Design (CAD) of Integrated Circuits and Systems*, 19(9):969–980, Sept. 2000.

[16] D. BLAUUW, S. SIRICHOTIYAKUL, and C. OH. Driver Modeling and Alignment for Worst-Case Delay Noise. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 11(2):157–166, Apr. 2003.

[17] S. BOBBA, I. N. HAJJ, and N. R. SHANBHAG. Analytical Expressions for Average Bit Statistics of Signal Lines in DSP Architectures. In *Intl. Symp. on Circuits and Systems (ISCAS)*, pages 33–36, Monterey, California, June 1998.

[18] A. BOCCA, S. SALERNO, E. MACII, and M. PONCINO. Energy-efficient Bus Encoding for LCD Displays. In *Great Lakes Symp. on VLSI (GLSVLSI)*, pages 240–243, Boston, Massachusetts, Apr. 2004.

[19] K. D. BOESE, A. B. KAHNG, and G. ROBINS. High-Performance Routing Trees With Identified Critical Sinks. In *Design Automation Conf. (DAC)*, pages 182–187, Dallas, Texas, June 1993.

[20] E. BOGATIN. *Signal Integrity - Simplified*. Prentice Hall, Upper Saddle River, New Jersey, 2004.

[21] K. BONDALAPATI and V. K. PRASANNA. Reconfigurable Computing Systems. *Proceedings of the IEEE*, 90(7):1201–1217, July 2002.

[22] S. BORKAR, T. KARNIK, S. NARENDRA, J. TSCHANZ, A. KESHAVARZI, and V. DE. Parameter Variations and Impact on Circuits and Microarchitecture. In *Design Automation Conf. (DAC)*, pages 338–342, Anaheim, California, June 2003.

[23] R. A. BRUALDI and A. J. HOFFMAN. On the Spectral Radius of (0,1)-Matrices. *Linear Algebra and Its Applications (Elsevier)*, 65:133–146, 1985.

[24] R. E. BRYANT, K.-T. CHENG, A. B. KAHNG, K. KEUTZER, W. MALY, R. NEWTON, L. PILEGGI, J. M. RABAEY, and A. SANGIOVANNI-VINCENTELLI. Limitations and Challenges of Computer-Aided Design Technology for CMOS VLSI. *Proceedings of the IEEE*, 89(3):341–365, Mar. 2001.

[25] Y. CAO, X. HUANG, N. H. CHANG, S. LIN, O. S. NAKAGAWA, W. XIE, D. SYLVESTER, and C. HU. Effective On-Chip Inductance Modeling for Multiple Signal Lines and Application to Repeater Insertion. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 10(6):799–805, Dec. 2002.

[26] M. CELIK, L. PILEGGI, and A. ODABASIOGLU. *IC Interconnect Analysis*. Kluwer, Norwell, Massachusetts, 1996.

[27] A. CHAKRABORTY, E. MACII, and M. PONCINO. Exploiting Cross-Channel Correlation for Energy-Efficient LCD Bus Encoding. In *Intl. Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, pages 297–307, Leuven, Belgium, Sept. 2005.

[28] A. P. CHANDRAKASAN and R. W. BRODERSEN. *Low Power Digital CMOS Design*. Kluwer, Norwell, Massachusetts, 1995.

[29] J. CHEN and L. HE. Determination of Worst-Case Crosstalk Noise for Non-Switching Victims in GHz+ Interconnects. In *Asia and South Pacific Design Automation Conf. (ASPDAC)*, pages 162–167, Kitakyushu, Japan, Jan. 2003.

[30] C.-K. CHENG, J. LILLIS, S. LIN, and N. CHANG. *Interconnect Analysis and Synthesis*. John Wiley & Sons, New York City, 2000.

[31] C. CHU and D. F. WONG. Closed Form Solution to Simultaneous Buffer Insertion/Sizing and Wire Sizing. *ACM Trans. on Design Automation of Electronic Systems*, 6(3):343–371, July 2001.

[32] J. CONG. An Interconnect-Centric Design Flow for Nanometer Technologies. *Proceedings of the IEEE*, 89(4):505–528, Apr. 2001.

[33] J. CONG, Y. FAN, G. HAN, X. YANG, and Z. ZHANG. Architecture and Synthesis for On-chip Multi-cycle Communication. *IEEE Trans. on Computer-Aided Design (CAD) of Integrated Circuits and Systems*, 23(4):550–564, Apr. 2004.

[34] J. CONG, K. S. LEUNG, and D. ZHOU. Performance-Driven Interconnect Design Based on Distributed RC Delay Model. In *Design Automation Conf. (DAC)*, pages 606–611, 1993.

[35] G. CONSTANTINE. Lower Bounds on the Spectra of Symmetric Matrices with Nonnegative Entries. *Linear Algebra and Its Applications (Elsevier)*, 65:171–178, 1985.

[36] F. DARTU, N. MENEZES, and L. T. PILEGGI. Performance Computation for Precharacterized CMOS Gates with RC Loads. *IEEE Trans. on Computer-Aided Design (CAD) of Integrated Circuits and Systems*, 15(5):544–553, May 1996.

[37] J. A. DAVIS and J. D. MEINDL, eds. *Interconnect Technology and Design for Gigascale Integration*. Kluwer, Norwell, Massachusetts, 2003.

[38] R. DEOKAR. Delay Calculation Meets the Nanometer Era. Technical paper, Cadence, 2004. (EE Times Online, Apr. 2004).

[39] A. DEUTSCH, P. W. COTEUS, G. V. KOPCSAY, H. H. SMITH, C. W. SUROVIC, B. L. KRAUTER, D. C. EDELSTEIN, and P. J. RESTLE. On-Chip Wiring Design Challenges for Gigahertz Operation. *Proceedings of the IEEE*, 89(5):529–555, May 2001.

[40] DEVICE RESEARCH GROUP, UC BERKELEY. BSIM 4.5.0 Release. http://www-device.eecs.berkeley.edu/bsim3, Feb. 2006.

[41] M. A. EL-MOURSY and E. G. FRIEDMAN. *Interconnect-Centric Design for Advanced SoC and NoC*, chapter 4 - Design Methodologies for On-Chip Inductive Interconnects, pages 85–124. Kluwer, Dordrecht, The Netherlands, 2004.

[42] W. C. ELMORE. The Transient Analysis of Damped Linear Networks with Particular Regard to Wideband Amplifiers. *Journal of Applied Physics*, 18(1):55–63, 1948.

[43] Y. EO. *Layout Optimization in VLSI Design*, chapter Modeling and Characterization of IC Interconnects and Packagings for the Signal Integrity Verification on High-Performance VLSI Circuits, pages 155–190. Kluwer, Dordrecht, The Netherlands, 2001.

[44] Y. EO and W. R. EISENSTADT. Simulation and Measurement of Picosecond Signal Transients, Propagation, and Crosstalk on Lossy VLSI Interconnect. *IEEE Trans. on Components, Packaging, and Manufacturing Technology*, 18(1):215–225, Mar. 1995.

[45] A. FORESTIER and M. STAN. Limits to Voltage Scaling from the Low Power Perspective. In *Intl. Symp. on Integrated Circuits and Systems Design*, pages 365–370, Manaus, Brazil, Sept. 2000.

[46] W. FORNACIARI, D. SCIUTO, and C. SILVANO. Power Estimation for Architectural Exploration of HW/SW Communication on System-Level Buses. In *Intl. Workshop on Hardware/Software Codesign (CODES)*, pages 152–156, Rome, Italy, May 1999.

[47] S. FRIEDLAND. The Maximal Eigenvalue of 0-1 Matrices with Prescribed Number of Ones. *Linear Algebra and Its Applications (Elsevier)*, 69:33–69, 1985.

[48] S. FRIEDLAND. Bounds on the Spectral Radius of Graphs with e Edges. *Linear Algebra and Its Applications (Elsevier)*, 101:81–86, 1988.

[49] S. FRIEDLAND. Lower Bounds for the First Eigenvalues of Certain M-Matrices Associated with Graphs. *Linear Algebra and Its Applications (Elsevier)*, 172:71–84, 1992.

[50] K. GALA, D. BLAAUW, J. WANG, V. ZOLOTOV, and M. ZHAO. Inductance 101: Analysis and Design Issues. In *Design Automation Conf. (DAC)*, pages 329–334, Las Vegas, Nevada, June 2001.

[51] K. GALA, V. ZOLOTOV, R. PANDA, B. YOUNG, J. WANG, and D. BLAAUW. On-Chip Inductance Modeling and Analysis. In *Design Automation Conf. (DAC)*, pages 63–68, Los Angeles, California, June 2000.

[52] A. GARCÍA ORTIZ. *Stochastic Data Models for Power Estimation at High-Levels of Abstraction*. PhD thesis, Darmstadt Univ. of Technology, Germany, 2003. Shaker.

[53] L. GARGANO, J. KÖRNER, and U. VACCARO. Sperner Capacities. *Graphs and Combinatorics*, 9:31–46, 1993.

[54] S. H. GEREZ. *Algorithms for VLSI Design Automation*. Wiley, Chichester, England, 1999.

[55] R. GOERING. Inductance Nags, Tools Lag in 0.13-Micron ICs. http://eetimes.com, Sept. 28th 2001. EE Times.

[56] R. GOERING. IC Buffering Panel Pits "Chickens" vs. "Ostriches". http://eedesign.com, Apr. 21st 2004. EE Design.

[57] F. W. GROVER. *Inductance Calculations. Working Formulas and Tables*. Dover Publications, Mineola, New York, 1946.

[58] L. HE. *Layout Optimization in VLSI Design*, chapter Interconnect Modeling and Design With Consideration of Inductance, pages 155–190. Kluwer, Dordrecht, The Netherlands, 2001.

[59] L. HE, N. CHANG, S. LIN, and O. S. NAKAGAWA. An Efficient Inductance Modeling for On-Chip Interconnects. In *IEEE Custom Integrated Circuits Conf.*, pages 457–460, San Diego, California, May 1999.

[60] K. HIROSE and H. YASUURA. A Bus Delay Reduction Technique Considering Crosstalk. In *Design Automation and Test in Europe (DATE)*, pages 441–445, Paris, France, Mar. 2000.

[61] C. W. HO, D. A. CHANCE, C. H. BAJOREK, and R. E. ACOSTA. The Thin-Film Module as a High-Performance Semiconductor Package. *IBM Journal on Research and Development*, 26(3):286–296, May 1982.

[62] D. A. HODGES, H. G. JACKSON, and R. A. SALEH. *Analysis and Design of Digital Integrated Circuits in Deep Submicron Technology*. McGraw-Hill, New York City, 2003.

[63] S. HONG, U. NARAYANAN, K.-S. CHUNG, and T. KIM. Bus-Invert Coding for Low-Power I/O - A Decomposition Approach. In *IEEE Midwest Symp. on Circuits and Systems*, pages 750–753, Lansing, Michigan, Aug. 2000.

[64] T. INDERMAUR and M. HOROWITZ. Evaluation of Charge Recovery Circuits and Adiabatic Switching for Low Power CMOS Design. In *IEEE Symp. on Low Power Electronics*, pages 102–103, San Diego, California, Oct. 1994.

[65] INTEL CORP. A Prediction Made Real Improves Billions of Lives. http://www.intel.com/technology/silicon/mooreslaw, Apr. 2006.

[66] INTERNATIONAL TECHNOLOGY ROADMAP FOR SEMICONDUCTORS, 2005 EDITION. *Executive Summary*. http://www.itrs.net, Mar. 2006.

[67] INTERNATIONAL TECHNOLOGY ROADMAP FOR SEMICONDUCTORS, 2005 EDITION. *Interconnect*. http://www.itrs.net, Mar. 2006.

[68] A. M. IONESCU and K. BANERJEE, eds. *Emerging Nanoelectronics. Life After CMOS*, volume 1, 2, and 3. Kluwer, Norwell, Massachusetts, 2005.

[69] Y. ISMAIL, E. G. FRIEDMANN, and J. L. NEVES. Figures of Merit to Characterize the Importance of On-Chip Inductance. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 7:442–449, Dec. 1999.

[70] Y. I. ISMAIL and E. G. FRIEDMAN. *On-Chip Inductance in High Speed Integrated Circuits*. Kluwer, Norwell, Massachusetts, 2001.

[71] Y. I. ISMAIL, E. G. FRIEDMAN, and J. L. NEVES. Equivalent Elmore Delay for RLC Trees. In *Design Automation Conf. (DAC)*, pages 715–720, New Orleans, Louisiana, June 1999.

[72] A. JANTSCH and H. TENHUNEN, eds. *Networks on Chip*. Kluwer, Hingham, Massachusetts, 2003.

[73] W. JIN, Y. EO, W. R. EISENSTADT, and J. SHIM. Fast and Accurate Quasi-Three-Dimensional Capacitance Determination of Multilayer VLSI Interconnects. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 9(3):450–460, June 2001.

[74] M. KAMON, N. A. MARQUES, L. M. SILVEIRA, and J. WHITE. Automatic Generation of Accurate Circuit Models of 3-D Interconnect. *IEEE Trans. on Components, Packaging, and Manufacturing Technology - Part B*, 21(3):225–240, Aug. 1998.

[75] M. KAMON, M. TSUK, and J. WHITE. FastHenry: A Multipole-Accelerated 3D Inductance Extraction Program. *IEEE Trans. on Microwave Theory and Techniques*, 42(9):1750–1758, Sept. 1994.

[76] C. V. KASHYAP, C. J. ALPERT, and A. DEVGAN. An Effective Capacitance Based Delay Metric for RC Interconnect. In *Intl. Conf. on Computer-Aided Design (ICCAD)*, pages 229–234, San Jose, California, Nov. 2000.

[77] C. V. KASHYAP and B. L. KRAUTER. A Realizable Driving Point Model for On-Chip Interconnect with Inductance. In *Design Automation Conf. (DAC)*, pages 190–195, Los Angeles, California, June 2000.

[78] H. KAUL, D. SYLVESTER, and D. BLAAUW. Active Shields: A New Approach to Shielding Global Wires. In *Great Lakes Symp. on VLSI (GLSVLSI)*, pages 112–117, New York, Apr. 2002.

[79] H. KAUL, D. SYLVESTER, and D. BLAAUW. Performance Optimization of Critical Nets Through Active Shielding. *IEEE Trans. on Circuits and Systems I: Regular Papers*, 51(12):2417–2435, Dec. 2004.

[80] K. KEUTZER, S. MALIK, R. NEWTON, J. RABAEY, and A. SANGIOVANNI-VINCENTELLI. System Level Design: Orthogonalization of Concerns and Platform-Based Design. *IEEE Trans. on Computer-Aided Design (CAD) of Integrated Circuits and Systems*, 19(12):1523–1543, Dec. 2000.

[81] Z. KHAN, A. T.ERDOGAN, and T. ARSLAN. Dual Low-Power and Crosstalk Immune Encoding Scheme for On-chip Data Buses. *Electronics Letters*, 39(20):1436–1437, Oct. 2003.

[82] A. KHANDEKAR, R. MCELIECE, and E. RODEMICH. *Coding, Communications, and Broadcasting*, chapter The Discrete Noiseless Channel Revisited, pages 115–137. Research Studies Press Ltd., Baldock, Hertfordshore, England, 2000. (Proc. of Intl. Symp. on Communications Theory and Applications, July 1999).

[83] K.-W. KIM, K.-H. BAEK, N. SHANBHAG, C. LIU, and S.-M. KANG. Coupling-Driven Signal Encoding Scheme for Low-Power Interface Design. In *Intl. Conf. on Computer-Aided Design (ICCAD)*, pages 318–321, San Jose, California, Nov. 2000.

[84] S. KIRKPATRICK, J. C. D. GELATT, and M. P. VECCHI. Optimization by Simulated Annealing. *Science*, 220(4598):671–680, May 1983.

[85] S. KOMATSU, M. IKEDA, and K. ASADA. Bus Data Encoding with Coupling-driven Adaptive Code-book Method for Low Power Data Transmission. In *European Solid-State Circuits Conf. (ESSCIRC)*, pages 312–315, Villach, Austria, Sept. 2001.

[86] T. K. KONSTANTAKOPOULOS. Implementation of Delay and Power Reduction in Deep Sub-Micron Buses Using Coding. Master's thesis, Massachusetts Inst. of Technology, May 2002.

[87] C. KRETZSCHMAR. *Verlustleistungsreduktion durch transitionsmindernde Kodierung von Systembussen*. PhD thesis, Technical Univ. of Chemnitz, Germany, 2006.

[88] C. KRETZSCHMAR, R. SIEGMUND, and D. MLLER. Adaptive Bus Encoding Technique for Switching Activity Reduced Data Transfer over Wide System Buses. In *Intl. Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, pages 66–75, Göttingen, Germany, Sept. 2000.

[89] C. KRETZSCHMAR, R. SIEGMUND, and D. MLLER. A Low Overhead Auto-optimizing Bus Encoding Scheme for Low Power Data Transmission. In *Intl. Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, pages 342–352, Seville, Spain, Sept. 2002.

[90] C. KRETZSCHMAR, R. SIEGMUND, and D. MLLER. Low Power Encoding Techniques for Dynamically Reconfigurable Hardware. *The Journal of Supercomputing*, 26(2):185–203, 2003.

[91] B. J. LAMERES and S. P. KHATRI. Bus Stuttering: An Encoding Technique to Reduce Inductive Noise in Off-Chip Data Transmission. In *Design Automation and Test in Europe (DATE)*, pages 522–527, Munich, Germany, Mar. 2006.

[92] M. LAMPROPOULOS, B. M. AL-HASHIMI, and P. M. ROSINGER. Minimization of Crosstalk Noise, Delay and Power Using a Modified Bus Invert Technique. In *Design Automation and Test in Europe (DATE)*, pages 1372–1373, Paris, France, Feb. 2004.

[93] P. E. LANDMAN and J. M. RABAEY. Power Estimation for High-level Synthesis. In *European Conf. on Design Automation with the European Event in ASIC Design*, pages 22–25, Paris, France, Feb. 1993.

[94] P. E. LANDMAN and J. M. RABAEY. Architectural Power Analysis: The Dual Bit Type Model. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 3(2):173–187, June 1995.

[95] H. LEKATSAS and J. HENKEL. ETAM++: Extended Transition Activity Measure for Low Power Address Bus Designs. In *Asia and South Pacific Design Automation Conf. (ASPDAC)*, pages 113–120, Bangalore, India, Jan. 2002.

[96] R. LI, D. ZHOU, J. LIU, and X. ZENG. Power-Optimal Simultaneous Buffer Insertion/Sizing and Wire Sizing. In *Intl. Conf. on Computer-Aided Design (ICCAD)*, pages 581–586, San, Jose, California, Nov. 2003.

[97] J. LILLIS, C.-K. CHENG, and T.-T. LIN. Optimal Wire Sizing and Buffer Insertion for Low Power and a Generalized Delay Model. *IEEE Journal of Solid-State Circuits*, 31(4):437–447, Apr. 1996.

[98] R.-B. LIN. Coupling Reduction Analysis of Bus-Invert Coding. In *Intl. Symp. on Circuits and Systems (ISCAS)*, volume 6, pages 5862–5865, Kobe, Japan, May 2005.

[99] R.-B. LIN and C.-M. TAI. Theoretical Analysis of Bus-Invert Coding. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 10(6):929–935, Dec. 2002.

[100] S. LIN, N. CHANG, and S. NAKAGAWA. Quick On-Chip Self- and Mutual-Inductance Screen. In *Intl. Symp. on Quality Electronic Design*, pages 513–520, San Jose, California, Mar. 2000.

[101] J. LÖFVENBERG. Non-Redundant Coding for Deep Sub-Micron Address Buses. In *Proc. of 4th IEEE Intl. Workshop on System-on-Chip for Real-Time Applications (IWSOC'04)*, pages 275–279, Banff, Alberta, Canada, July 2004.

[102] L. LOVÁSZ. On the Shannon Capacity of a Graph. *IEEE Trans. on Information Theory*, 25(1):1–7, Jan. 1979.

[103] B. LU, D.-Z. DU, and S. S. SAPATNEKAR, eds. *Layout Optimization in VLSI Design*. Kluwer, Dordrecht, The Netherlands, 2001.

[104] T. LV, J. HENKEL, H. LEKATSAS, and W. WOLF. A Dictionary-Based En/Decoding Scheme for Low-Power Data Buses. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 11(5):943–951, Oct. 2003.

[105] C.-G. LYUH, T. KIM, and K.-W. KIM. Coupling-Aware High-level Interconnect Synthesis for Low Power. In *Intl. Conf. on Computer-Aided Design (ICCAD)*, pages 609–613, San Jose, California, Nov. 2002.

[106] Y. MA, S. DONG, S. CHEN, C. K. CHENG, and J. GU. Buffer Planning as an Integral Part of Floorplanning With Consideration of Routing Congestion. *IEEE Trans. on Computer-Aided Design (CAD) of Integrated Circuits and Systems*, 24:609–621, Apr. 2005.

[107] L. MACCHIARULO, E. MACII, and M. PONCINO. Low-Energy Encoding for Deep-Submicron Address Buses. In *Intl. Symp. on Low Power Electronics and Design (ISLPED)*, pages 176–181, Huntington Beach, California, Aug. 2001.

[108] M. MAMIDIPAKA, D. HIRSCHBERG, and N. DUTT. Low Power Address Encoding Using Self-Organizing Lists. In *Intl. Symp. on Low Power Electronics and Design (ISLPED)*, pages 188–193, Huntington Beach, California, Aug. 2001.

[109] Y. MASSOUD and Y. ISMAIL. Grasping the Impact of On-Chip Inductance. *IEEE Circuits & Devices Magazine*, 17(4):14–21, July 2001.

[110] Y. MASSOUD, J. KAWA, D. MACMILLEN, and J. WHITE. Modeling and Analysis of Differential Signaling for Minimizing Inductive Cross-talk. In *Design Automation Conf. (DAC)*, pages 804–809, Las Vegas, Nevada, June 2001.

[111] Y. MASSOUD, S. MAJORS, T. BUSTAMI, and J. WHITE. Layout Techniques for Minimizing On-Chip Interconnect Self Inductance. In *Design Automation Conf. (DAC)*, pages 566–571, San Francisco, California, June 1998.

[112] Y. MASSOUD, S. MAJORS, J. KAWA, T. BUSTAMI, D. MACMILLEN, and J. WHITE. Managing On-Chip Inductive Effects. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 10(6):789–798, Dec. 2002.

[113] J. D. MEINDL. Beyond Moore's Law: The Interconnect Era. *Computing in Science & Engineering*, 5(1):20–24, Jan.–Feb. 2003.

[114] A. V. MEZHIBA and E. G. FRIEDMAN. Inductive Characteristics of Power Distribution Grids in High Speed Integrated Circuits. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 10(6):762–776, Dec. 2002.

[115] A. V. MEZHIBA and E. G. FRIEDMAN. Properties of On-Chip Inductive Current Loops. In *Great Lakes Symp. on VLSI (GLSVLSI)*, pages 12–17, New York City, Apr. 2002.

[116] A. V. MEZHIBA and E. G. FRIEDMAN. *Power Distribution Networks in High Speed Integrated Circuits*. Kluwer, Norwell, Massachusetts, 2004.

[117] F. MOLL and M. ROCA. *Interconnection Noise in VLSI Circuits*. Kluwer, Dordrecht, The Netherlands, 2004.

[118] G. E. MOORE. Cramming More Components onto Integrated Circuits. *Electronics Magazine*, 19 Apr. 1965.

[119] G. E. MOORE. Progress in Digital Integrated Electronics. In *Intl. Electron Device Meeting (IEDM)*, pages 11–13, Washington D.C., Dec. 1975.

[120] G. E. MOORE. No Exponential is Forever... but We Can Delay 'Forever'. In *Intl. Solid-State Circuits Conf. (ISSCC)*, San Francisco, California, Feb. 2003. Plenary Presentation.

[121] A.-T. MURGAN. *Principiile Teoriei Informaţiei în Ingineria Informaţiei şi a Comunicaţiilor*. Editura Academiei Române, Bucharest, Romania, 1998.

[122] E. MUSOLL, T. LANG, and J. CORTADELLA. Working-Zone Encoding for Reducing the Energy in Microprocessor Address Buses. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 6(4):568–572, Dec. 1998.

[123] K. NABORS and J. WHITE. FastCap: A Multipole-Accelerated 3D Capacitance Extraction Program. *IEEE Trans. on Computer-Aided Design (CAD) of Integrated Circuits and Systems*, 10(11):1447–1459, Nov. 1991.

[124] A. NALAMALPU, S. SRINIVASAN, and W. P. BURLESON. Boosters for Driving Long On-chip Interconnects – Design Issues, Interconnect Synthesis, and Comparison With Repeaters. *IEEE Trans. on Computer-Aided Design (CAD) of Integrated Circuits and Systems*, 21(1):50–62, Jan. 2002.

[125] NANOSCALE INTEGRATION AND MODELING (NIMO) GROUP, ARIZONA STATE UNIV. Predictive Technology Model. http://www.eas.asu.edu/ptm, Feb. 2006.

[126] U. NARAYANAN, K.-S. CHUNG, and T. KIM. Enhanced Bus Invert Encodings for Low-Power. In *Intl. Symp. on Circuits and Systems (ISCAS)*, volume V, pages 25–28, Scottsdale, Arizona, May 2002.

[127] S. G. NARENDRA and A. CHANDRAKASAN, eds. *Leakage in Nanometer CMOS Technologies*. Springer, New York City, 2006.

[128] W. NEBEL and J. MERMET, eds. *Low Power Design in Deep Submicron Electronics*. Kluwer, Dordrecht, The Netherlands, 1997.

[129] A. NOURRACHMAT, S. SALERNO, E. MACII, and M. PONCINO. Energy-Efficient Color Approximation for Digital LCD Interfaces. In *Intl. Conf. on Computer Design (ICCD)*, pages 81–86, San Jose, California, Oct. 2005.

[130] P. R. O'BRIEN and T. L. SAVARINO. Modeling the Driving Point Characteristic of Resistive Interconnect for Accurate Delay Estimation. In *Intl. Conf. on Computer-Aided Design (ICCAD)*, pages 512–515, San Jose, California, Nov. 1989.

[131] T. OKAMOTO and J. CONG. Buffered Steiner Tree Cnstruction with Wire Sizing for Interconnect Layout Optimization. In *Intl. Conf. on Computer-Aided Design (ICCAD)*, pages 44–49, San Jose, California, Nov. 1996.

[132] R. M. OWENS, H. MEHTA, and M. J. IRWIN. Some Issues in Gray Coding. In *Great Lakes Symp. on VLSI (GLSVLSI)*, pages 170–180, Ames, Iowa, Mar. 1996.

[133] D. PAMUNUWA. *Modelling and Analysis of Interconnects for Deep Submicron Systems-on-Chip*. PhD thesis, Royal Inst. of Technology, Stockholm, Sweden, 2003.

[134] S. PANDEY and M. GLESNER. Energy Efficient MPSoC On-Chip Communication Bus Synthesis Using Voltage Scaling Technique. In *Intl. Symp. on Circuits and Systems (ISCAS)*, Island of Kos, Greece, May 2006.

[135] S. PANDEY and M. GLESNER. Statistical On-Chip Communication Bus Synthesis and Voltage Scaling Under Timing Yield Constraint. In *Design Automation Conf. (DAC)*, San Francisco, California, July 2006.

[136] A. PAPOULIS. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, Singapore, 3rd edition, 1991.

[137] C. R. PAUL. *Analysis of Multiconductor Transmission Lines*. John Wiley & Sons, New York City, 1994.

[138] M. PEDRAM and J. RABAEY, eds. *Power Aware Design Methodologies*. Kluwer, Norwell, Massachusetts, 2002.

[139] L. T. PILLAGE and R. A. ROHRER. Asymptotic Waveform Evaluation for Timing Analysis. *IEEE Trans. on Computer-Aided Design (CAD) of Integrated Circuits and Systems*, 9(4):352–366, Apr. 1990.

[140] J. QIAN, S. PULLELA, and L. PILLAGE. Modeling the Effective Capacitance for the RC Interconnect of CMOS Gates. *IEEE Trans. on Computer-Aided Design (CAD) of Integrated Circuits and Systems*, 13(12):1526–1535, Dec. 1994.

[141] J. M. RABAEY, A. CHANDRAKASAN, and B. NIKOLIĆ. *Digital Integrated Circuits. A Design Perspective*. Prentice Hall, Upper Saddle River, New Jersey, 2nd edition, 2003.

[142] J. M. RABAEY and M. PEDRAM, eds. *Low Power Design Methodologies*. Kluwer, Norwell, Massachusetts, 1996.

[143] S. RAMPRASAD, N. R. SHANBAG, and I. N. HAJJ. A Coding Framework for Low-Power Address and Data Busses. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 7(2):212–221, June 1999.

[144] S. RAMPRASAD, N. R. SHANBAG, and I. N. HAJJ. Signal Coding for Low-Power: Fundamental Limits and Practical Realizations. *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, 46(7):923–929, July 1999.

[145] S. RAMPRASAD, N. R. SHANBHAG, and I. N. HAJJ. Analytical Estimation of Signal Transition Activity from Word-Level Statistics. *IEEE Trans. on Computer-Aided Design (CAD) of Integrated Circuits and Systems*, 16(7):718–733, July 1997.

[146] S. RAMPRASAD, N. R. SHANBHAG, and I. N. HAJJ. Analytical Estimation of Transition Activity From Word-Level Signal Statistics. In *Design Automation Conf. (DAC)*, pages 582–587, Anaheim, California, June 1997.

[147] R. R. RAO, H. S. DEOGUN, D. BLAAUW, and D. SYLVESTER. Bus Encoding for Total Power Reduction Using a Leakage-Aware Buffer Configuration. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 13(12):1376–1383, Dec. 2005.

[148] S. K. RAO, P. SADAYAPPAN, F. K. HWANG, and P. W. SHORT. The Rectilinear Steiner Arborescence Problem. *Algorithmica*, 7:277–288, 1992.

[149] E. ROGAI. *Tabele şi Formule Matematice*. Editura Tehnică, 1983.

[150] E. B. ROSA. The Self and Mutual Inductance of Linear Conductors. *Bulletin of the Bureau of Standards*, 4(2):301–344, Jan. 1908.

[151] E. B. ROSA and L. COHEN. Formulæand Tables for the Calculation of Mutual and Self-Inductance. *Bulletin of the Bureau of Standards*, 5(1):1–132, Aug. 1908.

[152] E. B. ROSA and F. W. GROVER. Formulæand Tables for the Calculation of Mutual and Self-Inductance. *Bulletin of the Bureau of Standards*, 8(1):1–237, Aug. 1912.

[153] K. ROY and S. C. PRASAD. *Low-Power CMOS VLSI Circuit Design*. John Wiley & Sons, New York City, 1995.

[154] J. RUBINSTEIN, P. PENFIELD JR., and M. A. HOROWITZ. Signal Delay in RC Tree Networks. *IEEE Trans. on Computer-Aided Design (CAD) of Integrated Circuits and Systems*, 2(3):202–211, July 1983.

[155] A. E. RUEHLI. Inductance Calculation in a Complex Integrated Circuit Environment. *IBM Journal on Research and Development*, 16:470–481, Sept. 1972.

[156] A. E. RUEHLI and A. C. CANGELLARIS. Progress in the Methodologies for the Electrical Modeling of Interconnects and Electronic Packages. *Proceedings of the IEEE*, 89(5):740–771, 2001.

[157] A. E. RUEHLI and H. HEEB. Challenges and Advances in Electrical Interconnect Analysis. In *Design Automation Conf. (DAC)*, pages 460–465, Anaheim, California, June 1992.

[158] K. S. SAINARAYANAN, J. V. R. RAVINDRA, and M. B. SRINIVAS. Minimizing Simultaneous Switching Noise (SSN) using Modified Odd/Even Bus Invert Method. In *IEEE Intl. Workshop on Electronic Design, Test and Applications (DELTA)*, pages 336–339, Kuala Lumpur, Malaysia, Jan. 2006.

[159] S. M. SAIT and H. YOUSSEF. *VLSI Physical Design Automation. Theory and Practice*. McGraw-Hill, New York City, 1995.

[160] T. SAKURAI and K. TAMARU. Simple Formulas for Two- and Three-Dimensional Capacitances. *IEEE Trans. on Electron Devices*, 30(2):183–185, Feb. 1983.

[161] S. SALERNO, A. BOCCA, E. MACII, and M. PONCINO. Limited Intra-word Transition Codes: An Energy-efficient Bus Encoding for LCD Display Interfaces. In *Intl. Symp. on Low Power Electronics and Design (ISLPED)*, pages 206–211, Newport Beach, California, Aug. 2004.

[162] S. SALERNO, E. MACII, and M. PONCINO. A Low-Power Encoding Scheme for GigaByte Video Interfaces. In *Intl. Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, pages 58–68, Santorini, Greece, Sept. 2004.

[163] A. SANGIOVANNI VINCENTELLI and G. MARTIN. Platform-Based Design and Software Design Methodologies for Embedded Systems. *IEEE Design & Test of Computers*, pages 23–33, Nov.-Dec. 2001.

[164] S. SAPATNEKAR. *Timing*. Kluwer, Norwell, Massachusetts, 2004.

[165] P. SAXENA, N. MENEZES, P. COCCHINI, and D. A. KIRKPATRICK. Repeater Scaling and Its Impact on CAD. *IEEE Trans. on Computer-Aided Design (CAD) of Integrated Circuits and Systems*, 23(4):451–463, Apr. 2004.

[166] C. SCHLACHTA, B. VOSS, and M. GLESNER. A Low-Power Line Driver Using Resonant Charging. In *Design Automation and Test in Europe (DATE), Designer's Forum*, Paris, France, Mar. 2002.

[167] S. SHANMUGAN and A. BREIPOHL. *Random Signals: Detection, Estimation and Data Analysis*. John Wiley & Sons, New York City, 1998.

[168] C. E. SHANNON. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27:379–423, 623–656, July–Oct. 1948.

[169] K. L. SHEPARD and Z. TIAN. Return-Limited Inductances: A Practical Approach to On-Chip Inductance Extraction. *IEEE Trans. on Computer-Aided Design (CAD) of Integrated Circuits and Systems*, 19(4):425–436, Apr. 2000.

[170] N. A. SHERWANI. *Algorithms for VLSI Physical Design Automation*. Springer, New York City, 3rd edition, 2005.

[171] Y. SHIN, S.-I. CHAE, and K. CHOI. Partial Bus-Invert Coding for Power Optimization of Application-Specific Systems. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 9(2):377–383, Apr. 2001.

[172] Y. SHIN, K. CHOI, and Y.-H. CHANG. Narrow Bus Encoding for Low-Power DSP Systems. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 9(5):656–660, Oct. 2001.

[173] S. SIRICHOTIYAKUL, D. BLAUUW, C. OH, R. LEVY, and V. ZOLOTOV. Driver Modeling and Alignment for Worst-Case Delay Noise. In *Design Automation Conf. (DAC)*, pages 720–725, Las Vegas, Nevada, June 2001.

[174] J. SNELLMAN. The Maximal Spectral Radius of a Digraph with $(M+1)^2 - S$ Edges. *Electronic Journal of Linear Algebra*, 10:179–189, 2003. Corrigendum.

[175] P. P. SOTIRIADIS. *Interconnect Modeling and Optimization in Deep Sub-Micron Technologies*. PhD thesis, Massachusetts Inst. of Technology, May 2002.

[176] P. P. SOTIRIADIS. *Interconnect-Centric Design for Advanced SoC and NoC*, chapter Power Reduction Coding for Buses, pages 177–206. Kluwer, Dordrecht, The Netherlands, 2004.

[177] P. P. SOTIRIADIS and A. CHANDRAKASAN. Bus Energy Minimization by Transition Pattern Coding in Deep Sub-Micron Technologies. In *Intl. Conf. on Computer-Aided Design (ICCAD)*, pages 322–328, San Jose, California, Nov. 2000.

[178] P. P. SOTIRIADIS and A. CHANDRAKASAN. Bus Energy Reduction by Transition Pattern Coding Using a Detailed Deep Submicrometer Bus Model. *IEEE Trans. on Circuits and Systems I*, 50(10):1280–1295, Oct. 2003.

[179] P. P. SOTIRIADIS and A. P. CHANDRAKASAN. Reducing Bus Delay in Submicron Technology Using Coding. In *Asia and South Pacific Design Automation Conf. (ASPDAC)*, pages 109–114, Yokohama, Japan, Jan.-Feb. 2001.

[180] P. P. SOTIRIADIS and A. P. CHANDRAKASAN. A Bus Energy Model for Deep-Submicron Technology. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 10(3):341–350, June 2002.

[181] S. R. SRIDHARA, A. AHMED, and N. R. SHANBHAG. Area and Energy-Efficient Crosstalk Avoidance Codes for On-Chip Buses. In *Intl. Conf. on Computer Design (ICCD)*, pages 12–17, San Jose, California, Oct. 2004.

[182] A. SRIVASTAVA, D. SYLVESTER, and D. BLAAUW. Power Minimization Using Simultaneous Gate-Sizing, Dual-Vdd, and Dual-Vth Assignment. In *Design Automation Conf. (DAC)*, pages 783–787, San Diego, California, June 2004.

[183] A. SRIVASTAVA, D. SYLVESTER, and D. BLAAUW. *Statistical Analysis and Optimization for VLSI: Timing and Power*. Springer, New York City, 2005.

[184] A. SRIVASTAVA, D. SYLVESTER, D. BLAAUW, and A. AGARWAL. Statistical Optimization of Leakage Power Considering Process Variations Using Dual-Vth and Sizing. In *Design Automation Conf. (DAC)*, pages 773–778, San Diego, California, June 2004.

[185] M. R. STAN. Low Power Encoding for VLSI and ECC Duals. In *Intl. Symp. on Information Theory*, page 19, Boston, Masachussettes, Aug. 1998.

[186] M. R. STAN and W. P. BURLESON. Limited-Weight Codes for Low-Power I/O. In *Intl. Workshop on Low Power Design*, pages 209–214, Napa, California, Apr. 1994.

[187] M. R. STAN and W. P. BURLESON. Bus-Invert Coding for Low-Power I/O. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 3(1):49–58, Mar. 1995.

[188] M. R. STAN and W. P. BURLESON. Two-Dimensional Codes for Low-Power. In *Intl. Symp. on Low Power Electronics and Design (ISLPED)*, pages 335–340, Monterey, California, Aug. 1996.

[189] M. R. STAN and W. P. BURLESON. Low-Power Encodings for Global Communication in CMOS VLSI. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 5(4):444–455, Dec. 1997.

[190] M. R. STAN and Y. ZHANG. Perfect 3-Limited-Weight Code for Low Power I/O. In *Intl. Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, pages 79–89, Santorini, Greece, Sept. 2004.

[191] D. SYLVESTER and K. KEUTZER. Impact of Small Process Geometries on Microarchitectures in Systems on a Chip. *Proceedings of the IEEE*, 89(4):467–489, Apr. 2001.

[192] S.-W. TU, J.-Y. JOU, and Y.-W. CHANG. RLC Effects on Worst-Case Switching Pattern for On-Chip Buses. In *Intl. Symp. on Circuits and Systems (ISCAS)*, volume 2, pages 945–948, Vancouver, Canada, May 2004.

[193] S.-W. TU, J.-Y. JOU, and Y.-W. CHANG. RLC Coupling-aware Simulation for On-chip Buses and Their Encoding for Delay Reduction. In *Intl. Symp. on Circuits and Systems (ISCAS)*, volume 4, pages 4134–4137, Kobe, Japan, May 2005.

[194] L. K. VAKATI and J. WANG. A New Multi-Ramp Driver Model with RLC Interconnect Load. In *Intl. Symp. on Physical Design (ISPD)*, pages 170–175, Phoenix, Arizona, Apr. 2004.

[195] L. P. P. P. VAN GINNEKEN. Buffer Placement in Distributed RC-tree Networks for Minimal Elmore Delay. In *Intl. Symp. on Circuits and Systems (ISCAS)*, pages 865–868, New Orleans, Louisiana, May 1990.

[196] B. VICTOR and K. KEUTZER. Bus Encoding to Prevent Crosstalk Delay. In *Intl. Conf. on Computer-Aided Design (ICCAD)*, pages 57–69, San Jose, California, Nov. 2001.

[197] A. VLADIMIRESCU. *The SPICE Book*. John Wiley & Sons, New York City, 1994.

[198] B. VOSS. *Resonantes Umladen als Schaltungstechnik zur Verlustleistungsreduktion in digitalen CMOS-Schaltungen*. PhD thesis, Darmstadt Univ. of Technology, Germany, 2001. Shaker.

[199] N. H. E. WESTE and K. ESHRAGHIAN. *Principles of CMOS VLSI Design. A Systems Perspective*. Addison-Wesley, Reading, Massachusetts, 2nd edition, 1993.

[200] S.-C. WONG, G.-Y. LEE, and D.-J. MA. Modeling of Interconnect Capacitance, Delay, and Crosstalk in VLSI. *IEEE Trans. on Semiconductor Manufacturing*, 13:108–111, Feb. 2000.

[201] S.-C. WONG, T. G.-Y. LEE, D.-J. MA, and C.-J. CHAO. An Empirical Three-Dimensional Crossover Capacitance Model for Multilevel Interconnect VLSI Circuits. *IEEE Trans. on Semiconductor Manufacturing*, 13(2):219–227, May 2000.

[202] J. YANG, R. GUPTA, and C. ZHANG. Frequent Value Encoding for Low Power Data Buses. *ACM Trans. on Design Automation of Electronic Systems (DAES)*, 9(3):354–384, July 2004.

[203] B. YOUNG. *Digital Signal Integrity: Modeling and Simulation with Interconnects and Packages*. Prentice Hall, Upper Saddle River, New Jersey, 2001.

[204] H. ZHANG, V. GEORGE, and J. RABAEY. Low-swing On-chip Signaling Techniques: Effectiveness and Robustness. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 8(3):264–272, June 2000.

[205] H. ZHANG, V. PRABHU, V. GEORGE, M. WAN, M. BENES, A. ABNOUS, and J. M. RABAEY. A 1-V Heterogeneous Reconfigurable DSP IC for Wireless Baseband DigitalSignal Processing. *IEEE Journal of Solid-State Circuits*, 35(11):1697–1704, Nov. 2000.

[206] Y. ZHANG, J. LACH, K. SKADRON, and M. R. STAN. Odd/Even Bus Invert with Two-Phase Transfer for Buses with Coupling. In *Intl. Symp. on Low Power Electronics and Design (ISLPED)*, pages 80–83, Monterey, California, Aug. 2002.

List of Publications

[207] T. MURGAN, M. MOMENI, A. GARCÍA ORTIZ, and M. GLESNER. A High-Level Compact Pattern-Dependent Delay Model for High-Speed Point-to-Point Interconnects. In *Intl. Conf. on Computer-Aided Design (ICCAD)*, San Jose, California, Nov. 2006.

[208] T. MURGAN, P. B. BACINSCHI, A. GARCÍA ORTIZ, and M. GLESNER. Partial Bus-Invert Bus Encoding Schemes for Low-Power DSP Systems Considering Inter-Wire Capacitance. In *Intl. Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, volume 4148 of *Lecture Notes in Computer Science (LNCS)*, pages 169–180, Montpellier, France, Sep. 2006. Springer.

[209] T. MURGAN, A. GARCÍA ORTIZ, M. MOMENI, L. S. INDRUSIAK, M. GLESNER, and R. A. DA LUZ REIS. Timing and Power Consumption in High-Speed Very Deep Sub-Micron On-Chip Interconnects. *Journal of Integrated Circuits and Systems*, 1(3), 2006.

[210] T. MURGAN, O. MITEA, S. PANDEY, P. B. BACINSCHI, and M. GLESNER. Simultaneous Placement and Buffer Planning for Reduction of Power Consumption in Interconnects and Repeaters. In *IFIP Intl. Conf. on VLSI-SoC*, Nice, France, Oct. 2006.

[211] T. MURGAN and M. GLESNER. Limits of Switching Power Consumption in Encoded Interconnects. In *IEEE Intl. Symp. on Signal, Circuit and Systems (ISSCS)*, volume 1, pages 27–30, Iaşi, Romania, July 2005.

[212] T. MURGAN, A. GARCÍA ORTIZ, M. PETROV, and M. GLESNER. A Linear Model for High-Level Delay Estimation in VDSM On-Chip Interconnects. In *Intl. Symp. on Circuits and Systems (ISCAS)*, volume 2, pages 1078–1081, Kobe, Japan, May 2005.

[213] T. MURGAN, A. GARCÍA ORTIZ, C. SCHLACHTA, H. ZIMMER, M. PETROV, and M. GLESNER. On Timing and Power Consumption in Inductively Coupled On-Chip Interconnects. In *Intl. Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, volume 3254 of *Lecture Notes in Computer Science (LNCS)*, pages 819–828, Santorini, Greece, Sep. 2004. Springer.

[214] T. MURGAN, M. PETROV, A. GARCÍA ORTIZ, R. LUDEWIG, P. ZIPF, T. HOLLSTEIN, M. GLESNER, B. OELKRUG, and J. BRAKENSIEK. Evaluation and Run-Time Optimisation of On-Chip Communication Structures in Reconfigurable Architectures. In *Intl. Conf. on Field Programmable Logic and Applications (FPLA)*, volume 2778 of *Lecture Notes in Computer Science (LNCS)*, pages 1111–1114, Lisbon, Portugal, Sep. 2003. Springer.

[215] T. MURGAN, A. GARCÍA ORTIZ, M. PETROV, and M. GLESNER. A Stochastic Framework for Communication Architecture Evaluation in Networks-on-Chip. In *IEEE Intl. Symp. on Signal, Circuit and Systems (ISSCS)*, volume 1, pages 253–256, Iaşi, Romania, July 2003.

[216] T. MURGAN, C. SCHLACHTA, M. PETROV, L. S. INDRUSIAK, A. GARCÍA ORTIZ, M. GLESNER, and R. REIS. Accurate Capture of Timing Parameters in Inductively-Coupled On-Chip Interconnects. In *Intl. Symp. on Integrated Circuits and Systems Design (SBCCI)*, pages 117–122, Porto de Galinhas, Pernambuco, Brazil, Sep. 2004.

[217] T. MURGAN, A. GARCÍA ORTIZ, and M. GLESNER. High-level Estimation and Optimization of Power and Delay in Nano-scaled Interconnects. Invited Keynote Presentation at the 6th MARLOW Workshop, Budapest, Hungary, Apr. 2005.

[218] T. MURGAN, A. M. OBEID, A. GUNTORO, P. ZIPF, M. GLESNER, and U. HEINKEL. Design and Implementation of a Multi-Core Architecture for Overhead Processing in Optical Transport Networks. In *Workshop on Reconfigurable Communication-centric SoCs (ReCoSoC)*, Montpellier, France, June 2005.

[219] T. MURGAN, M. PETROV, M. MAJER, P. ZIPF, M. GLESNER, and U. HEINKEL. Flexible Overhead Processing Architectures for G.709 Optical Transport Networks. In *GI/ITG/GMM Workshop on "Methoden und Beschreibungssprachen zur Modellierung und Verifikation von Schaltungen und Systemen"*, Kaiserslautern, Germany, Feb. 2004.

[220] T. MURGAN, M. PETROV, M. MAJER, P. ZIPF, M. GLESNER, U. HEINKEL, J. PLEICKHARDT, and B. BLEISTEINER. Adaptive Architectures for an OTN Processor: Reducing Design Costs Through Reconfigurability and Multiprocessing. In *ACM Computing Frontiers Conf.*, pages 408–414, Ischia, Italy, Apr. 2004.

[221] T. MURGAN and R. RĂDESCU. Backtracking Algorithm Applied to Generate the RLL Dictionary for Magnetic Channels. *Scientific Bulletin of the "Politehnica" University of Bucharest, Electrical Engineering*, 63(3-4), 2000.

[222] T. MURGAN, R. RĂDESCU, and J.-M. BECKER. A Combinatorial Method for a RLL Code. In *Intl. Workshop on Algebraic and Combinatorial Coding Theory*, Bansko, Bulgaria, June 2000.

[223] R. DOGARU, T. MURGAN, and M. GLESNER. A Compact Solution for Voice Compression Based on Cellular Neural Networks. In *IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing*, Grado, Italy, June 2003.

[224] A. GARCÍA ORTIZ, L. D. KABULEPA, T. MURGAN, and M. GLESNER. Moment-Based Power Estimation in Very Deep Submicron Technologies. In *Intl. Conf. on Computer-Aided Design (ICCAD)*, pages 107–112, San Jose, California, Nov. 2003.

[225] A. GARCÍA ORTIZ, T. MURGAN, and M. GLESNER. Transition Activity Estimation for General Correlated Data Distributions. In *Intl. Conf. on VLSI Design*, pages 440–445, New Delhi, India, Jan. 2003.

[226] A. GARCÍA ORTIZ, T. MURGAN, and M. GLESNER. Moment-based Estimation of Switching Activity for Correlated Distributions. In *Intl. Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, volume 3254 of *Lecture Notes in Computer Science (LNCS)*, pages 859–868, Santorini, Greece, Sep. 2004. Springer.

[227] A. GARCÍA ORTIZ, T. MURGAN, L. S. INDRUSIAK, and M. GLESNER. Power Consumption in Point-to-Point Interconnect Architectures. In *Intl. Symp. on Integrated Circuits and Systems Design (SBCCI)*, pages 150–160, Porto Alegre, Rio Grande do Sul, Brazil, Sep. 2002.

[228] A. GARCÍA ORTIZ, T. MURGAN, L. D. KABULEPA, L. S. INDRUSIAK, and M. GLESNER. High-Level Estimation of Power Consumption in Point-to-Point Interconnect Architectures. *Journal of Integrated Circuits and Systems*, 1(1):23–31, March 2004.

[229] M. GLESNER, T. HOLLSTEIN, L. S. INDRUSIAK, P. ZIPF, T. PIONTECK, M. PETROV, H. ZIMMER, and T. MURGAN. Reconfigurable Platforms for Ubiquitous Computing. In *ACM Computing Frontiers Conf.*, pages 377–389, Ischia, Italy, April 2004.

[230] M. GLESNER, T. HOLLSTEIN, and T. MURGAN. System Design Challenges in Ubiquitous Computing Environments. In *Conf. on Electronics and Microsystem Technology*, Tallin, Estonia, Oct. 2004. (invited).

[231] M. GLESNER, T. HOLLSTEIN, and T. MURGAN. System Design Challenges in Ubiquitous Computing Environments. In *Intl. Conf. on Microelectronics (ICM)*, pages 11–14, Tunis, Tunisia, Dec. 2004. (invited).

[232] M. GLESNER and T. MURGAN. System Design and Integration in Pervasive Appliances. In *GI/GMM/ITG "Multi-Nature Systems" Workshop*, pages 1–4, Ilmenau, Germany, Sep. 2003. (invited).

[233] M. GLESNER, T. MURGAN, L. S. INDRUSIAK, M. PETROV, and S. PANDEY. System Design and Integration in Pervasive Appliances. In *Intl. Conf. on Microelectronics, Devices and Materials*, pages 97–108, Ptuj, Slovenia, Oct. 2003. (invited).

[234] M. GLESNER, T. MURGAN, L. S. INDRUSIAK, M. PETROV, and S. PANDEY. System Design and Integration in Pervasive Appliances. *Journal of Microelectronics, Electronic Components and Materials (MIDEM)*, 33(4):276–282, Oct.-Dec. 2003.

[235] M. GLESNER, H. HINKELMANN, T. HOLLSTEIN, L. S. INDRUSIAK, T. MURGAN, A. M. OBEID, M. PETROV, T. PIONTECK, and P. ZIPF. Reconfigurable Embedded Systems: An Application-Oriented Perspective on Architectures and Design Techniques. In *Intl. Workshop on Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS)*, volume 3553 of *Lecture Notes in Computer Science (LNCS)*, pages 12–21, Samos, Greece, July 2005. Springer.

[236] L. S. INDRUSIAK, T. MURGAN, M. GLESNER, and R. REIS. Consistency Control in Data-driven Design Automation Environments. In *IEEE Intl. Symp. on Signal, Circuit and Systems (ISSCS)*, volume 2, pages 629–632, Iaşi, Romania, July 2005.

[237] R. LUDEWIG, A. GARCÍA ORTIZ, T. MURGAN, and M. GLESNER. Power Estimation Based on Transition Activity Analysis with Architecture Precise Rapid Prototyping System. In *IEEE Intl. Workshop on Rapid System Prototyping (RSP)*, pages 138–143, Darmstadt, Germany, July 2002.

[238] R. LUDEWIG, A. GARCÍA ORTIZ, T. MURGAN, and M. GLESNER. Hardware Assisted Signal Activity Analysis for Power Estimation in Rapid Prototyped Systems. *Design Automation for Embedded Systems Journal*, 8(4):297–308, Kluwer 2003.

[239] R. LUDEWIG, A. GARCÍA ORTIZ, T. MURGAN, J.-J. OCAMPO HIDALGO, and M. GLESNER. Emulation of Analog Components for the Rapid Prototyping of Wireless Baseband Systems. In *IEEE Intl. Workshop on Rapid System Prototyping (RSP)*, pages 172–178, San Diego, California, June 2003.

[240] A. M. OBEID, T. MURGAN, A. TAADOU, and M. GLESNER. HW/SW Design and Realization of a Size-reconfigurable DCT Accelerator. In *IEEE 12th Intl. Conf. on Electronics, Circuits and Systems (ICECS)*, Gammarth, Tunisia, Dec. 2005.

[241] S. PANDEY, P. ZIPF, O. SOFFKE, M. PETROV, T. MURGAN, M. GLESNER, and M. MÜHLHÄUSER. An Infrastructure for Distributed Computing and Context Aware Computing. In *UbiComp 2003 Workshop: Multi-Device Interfaces for Ubiquitous Peripheral Interaction*, Seattle, Washington, Oct. 2003.

[242] S. PANDEY, T. MURGAN, and M. GLESNER. Energy Conscious Simultaneous Voltage Scaling and On-chip Communication Bus Synthesis. In *IFIP Intl. Conf. on VLSI-SoC*, Nice, France, Oct. 2006.

[243] M. PETROV, T. MURGAN, F. MAY, M. VORBACH, P. ZIPF, and M. GLESNER. The XPP Architecture and its Co-Simulation within the Simulink Environment. In *Intl. Conf. on Field Programmable Logic and Applications (FPLA)*, volume 3203 of *Lecture Notes in Computer Science (LNCS)*, pages 761–770, Antwerpen, Belgium, Aug.-Sep. 2004. Springer.

[244] M. PETROV, T. MURGAN, A. M. OBEID, C. CHIŢU, P. ZIPF, J. BRAKENSIEK, and M. GLESNER. Dynamic Power Optimization of the Trace-Back Process for the Viterbi Algorithm. In *Intl. Symp. on Circuits and Systems (ISCAS)*, volume 2, pages 721–724, Vancouver, Canada, May 2004.

[245] M. PETROV, T. MURGAN, P. ZIPF, and M. PETROV. Functional Modeling Techniques for a Wireless LAN OFDM Transceiver. In *Intl. Symp. on Circuits and Systems (ISCAS)*, volume 4, pages 3970–3972, Kobe, Japan, May 2005.

[246] M. PETROV, A. M. OBEID, T. MURGAN, P. ZIPF, J. BRAKENSIEK, B. OELKRUG, and M. GLESNER. An Adaptive Trace-Back Solution for State-Parallel Viterbi Decoders. In *IFIP Intl. Conference on Very Large Scale Integration (IFIP-VLSI)*, pages 167–174, Darmstadt, Germany, Dec. 2003.

[247] O. SOFFKE, P. ZIPF, T. MURGAN, and M. GLESNER. A Signal Theory Based Approach to the Statistical Analysis of Combinatorial Nanoelectronic Circuits. In *Design Automation and Test in Europe (DATE)*, pages 632–637, Munich, Germany, Feb. 2006.

Supervised Theses

[248] JUAN JESÚS VELASCO VELEZ. Extended Algorithms for Buffer Insertion in Capacitively and Inductively Coupled Interconnects and Analysis of Power Consumption. Diploma thesis, Mar. 2006.

[249] TOBIAS VOLLBERG. Präzise On-Chip Messung der durch Übersprechen erzeugten Spannungsspitzen. Master's thesis, March 2006. *Co-advised with Petru B. Bacinschi*.

[250] OLIVER MITEA. Verfahren zur Reduktion des Leistungsverbrauchs bei der Planung von Verbindungsstrukturen und Einfügen von Leistungstreibern. Studienarbeit, Feb. 2006. *Co-advised with Sujan Pandey*.

[251] MIGUEL GARCÍA ORTIZ. Design and Prototyping of a Power Line Modem. Diploma thesis, Sept. 2005.

[252] PETRU BOGDAN BACINSCHI. Organic Electronics: Device Modeling and Circuit Simulation. Diploma thesis, June 2005. *Co-advised with Thomas Hollstein*.

[253] FRANCISCO AZNAR BALLESTA. High-Level Analysis of Delay and Switching Power in Deep Sub-Micron Interconnects. Diploma thesis, May 2005.

[254] ISMAIL DEFLAOUI. Analyse und VHDL-Implementierung einer Hardware-Erweiterung zur Kodierungsbeschleunigung in einem Multiprozessorsystem. Studienarbeit, May 2005. *Co-advised with Peter Zipf*.

[255] MASSOUD MOMENI. Electrical Parameter Extraction for Distributed Models of On-Chip Interconnects. Diploma thesis, Sept. 2005. *Co-advised with Rolf Schuhmann*

[256] SEBASTIAN VOGEL. Closed Form Solution for Optimal Buffer Sizing. Study thesis, Sept. 2005. External Thesis at the University of Illinois at Urbana-Champaign.

[257] MURTHY PALLA. Methods for Detection of Unvalid Crosstalk Aggressor Configurations. Master's thesis, July 2005. External Thesis. *Co-advised with Klaus Koch*.

[258] ABDELOUAHID TAADOU. HW/SW Co-Design eines Ogg-Vorbis Players. Diplomarbeit, Apr. 2005. *Co-advised with Abdulfattah M. Obeid*.

[259] HUSSEIN CHOKR. Entwicklung einer rekonfigurierbaren seriellen FFT/DCT Architektur. Bachelorarbeit, Dec. 2004. *Co-advised with Abdulfattah M. Obeid*.

[260] RAJKUMAR METHUKU. Design and FPGA Implementation of a Multi-Core Architecture for ITU-T G.709 Overhead Processing. Master's thesis, Dec. 2004. *Co-advised with Mihail Petrov*.

[261] NING TU. Processor-based Overhead Processing in Optical Transport Networks. Diploma thesis, Oct. 2004.

[262] JORGE PINAZO DONOSO. Graphical User Interface for a Transition Activity Based Power Estimation Tool. Diploma thesis, Sept. 2004.

[263] MARIA-LAURA ICHIM. New Algorithms to Indoor Localization of a Mobile Phone. Diploma thesis, July 2004. *Co-advised with Matthias Rychetsky*.

[264] ALEXANDER WERTH. Entwicklung und Bewertung einer Multiprozessorarchitektur für Overheadverarbeitung in optischen Netzen. Diplomarbeit, Apr. 2004. *Co-advised with Mihail Petrov*.

[265] ZHAOMING DAI. Entwurf einer ATA Schnittstelle für den Wishbone Bus als synthetisierbares SystemC Model. Studienarbeit, Nov. 2003. *Co-advised with Mihail Petrov*.

[266] SHIDE WANG. Modelierung eines 32-bit Prozessor-Cores auf Transaktionsebene. Studienarbeit, Oct. 2003. *Co-advised with Mihail Petrov*.

[267] MATEUSZ MAJER. Evaluation of Reconfigurable Architectures for Overhead Processing in High-Speed Optical Transport Networks. Diploma thesis, Oct. 2003.

[268] THOMAS PFEIFFER. Simulation und Prototyping eines DLL-basierten Rake Receivers. Studienarbeit, Apr. 2002. *Co-advised with Ralf Ludewig*.

Index

Curriculum Vitæ

Tudor A. MURGAN

Personal Data:

Date of birth:	February 7th, 1978
Place of birth:	Bucharest, Romania

Academic Formation:

1984 - 1992	Primary and lower secondary school at the German school "Hermann Oberth" (currently "Johann Wolfgang Goethe") in Bucharest
1992 - 1996	Upper secondary school (*Liceu*) at the National College for Informatics "Tudor Vianu" in Bucharest Degree: upper secondary school leaving diploma (*Diplomă de Bacalaureat*) and programmer certificate (*Atestat de Programator*) University-entrance exam (*Examen de Admitere*)
1996 - 2001	Student at the Faculty of Electronics, Communications, and Information Technology, "Politehnica" University Bucharest Degree: diploma in engineering (*Inginer Diplomat)*
2001 - 2006	Ph.D. student, teaching and research assistant at the Institute of Microelectronic Systems, Darmstadt University of Technology, Darmstadt, Germany
2002 - 2004	Scholarship holder and graduate member of the Graduate College (*Graduiertenkolleg*) "System Integration of Ubiquitous Computing in Information Technology" (*Systemintegration für ubiquitäres Rechnen in der Informationstechnik*) funded by the German Research Foundation (*DFG*)

"Înţelepciunea unui mag mi-a povestit odată
de-un val prin care nu putem străbate cu privirea,
păienjeniş ce-ascunde pretutindeni firea,
de nu vedem nimic din ce-i aievea…"

Lucian Blaga